AD-A048 342 LTV AEROSPACE CORP DALLAS TEX VOUGHT SYSTEMS DIV F/G 15/7
SEATIDE ANALYSIS PROCESS. VOLUME IID. NAVAL ENGAGEMENT MODEL (N--ETC(U)
FEB 75 DAAB09-72-C-0062
UNCLASSIFIED VSD-00.1636-VOL-2D-REV-A NL

1 of 3
ADA048 342

B.S.

# SEATIDE ANALYSIS PROCESS

# VOLUME IID

# NAVAL ENGAGEMENT MODEL (NEM)

# APPENDIX N

REPORT NO. 00.1636
JANUARY 1974
(CONTRACT DAAB09-72-C-0062)

VOUGHT SYSTEMS DIVISION
LTV AEROSPACE CORPORATION
P.O. BOX 5907 · DALLAS, TEXAS 75222

UNCLASSIFIED

VMSC/SEATIDE/ 1106
ITEM 1 COPY NO. 4
FILE

SEATIDE

6

# SEATIDE ANALYSIS PROCESS.

1

# VOLUME IID.

# NAVAL ENGAGEMENT MODEL (NEM).

# APPENDIX N. Revision A.

14 VSD-
REPORT NO. 00.1636 – VOL-2D-REV-A

JANUARY 1974

(CONTRACT DAAB09-72-C-0062)

15

REVISION A – 20 FEB 1975

11

12 207p.

VOUGHT SYSTEMS DIVISION
LTV AEROSPACE CORPORATION
P.O. BOX 5907 · DALLAS, TEXAS 75222

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

UNCLASSIFIED

408116 lb

## FOREWORD

(U) This report was prepared by the Vought Systems Division, LTV Aerospace Corporation, P.O. Box 6267, Dallas, Texas 75222 under U. S. Army Electronics Command Contract DAAB09-72-C-0062. The work was initiated under the direction of Captain R. A. Dowd, USN and completed under Captain W. A. Greene, USN, Chief, Long Range Forecast Division, Directorate of Estimates, Defense Intelligence Agency (DIA-DE-1).

(U) Persons contributing to the development and testing of the procedure reported herein include:

DIA-DE-1

Capt. W. A. Greene, USN
Capt. R. A. Dowd, USN (Retired)
Capt. R. F. Weiss, USAF
Mr. R. E. McQuiston

ARPA

Cmdr. T. W. Hogan, USN

VSD

| | |
|---|---|
| Mr. J. S. Smith, Jr. | Chief Project Engineer, Special Projects |
| Dr. L. D. Gregory | Project Engineer, SEATIDE |
| Mr. J. R. Matthews | Models and Analysis |
| Mr. F. E. Dye, Jr. | Technologies (CM-CGSM) |
| Mr. R. K. McDonough | Models and Analysis (CM-CGSM) |
| Mr. R. E. Dyer | Models and Analysis (NEM) |
| Mr. G. G. Johnson | Aerodynamics |
| Dr. J. A. Bottorff | Propulsion |
| Mr. A. C. Morris | Electronics (Radar, Guidance) |
| Mr. H. R. Crow | Operations Research |
| Mr. G. H. Harris | Operations Research |
| Mr. G. S. McCorkle | Propulsion |
| Mr. L. D. Cardwell | Propulsion |

(U) This report has been prepared in the following volumes:

| Volume | Classification | Title |
|---|---|---|
| I | S | Summary |
| IIA | U | Naval Engagement Model (NEM) - Users Manual |
| IIB | U | NEM - Appendices A - I |
| IIC | S | NEM - Appendices J - M |
| IID | U | NEM - Appendix N |
| IIIA | U | Cruise Missile - Concept Generation and Screening Model (CM-CGSM) - Users Manual |
| IIIB | U | CM-CGSM Appendices A-B |
| IIIC | S | CM-CGSM Appendix C |
| IIID | U | CM-CGSM Appendices D-G |
| IIIE | U | CM-CGSM Appendix H |
| IV | S | Relative Worth Model (RWM) |
| V | U | Relative Cost Model (RCM) |

# ABSTRACT

(U) The SEATIDE Analysis Process is a semi-automated procedure for the generation of time-phased, high value cruise missile weapon systems concepts, together with the supporting technology and intelligence indicators which would reflect that these technological goals are being achieved. The SEATIDE process can also be used to evaluate the effectiveness of fixed force levels, existing forces in SAL environments, or Naval defenses.

(U) The Defense Intelligence Agency, through its Directorate of Estimates, and The Advanced Research Projects Agency (ARPA) have sponsored the development of this computer based analysis at the weapon system and Naval force structure level. A previous process, RIPTIDE, was developed for DIA for use in analysis of strategic missile systems.

(U) Generic to the SEATIDE Analysis Process are three major computer models: The Naval Engagement Model (NEM), Cruise Missile Concept Generation and Screening Model (CM-CGSM) and Relative Worth Model (RWM). The NEM evaluates force effectiveness, tactics, and task force configurations; the CM-CGSM enables definition and selection of candidate, advanced cruise missile system concepts; and the RWM permits assessment of worth in accordance with a variety of objective and subjective criteria. Each of these models has been checked out by DIA.

(U) In addition to exercising the computer models, there are several other analytical and engineering tasks to be performed, e.g., the identification of areas of current interest and the associated criteria and potential concepts, the creation of a foreign technology data bank in a format needed by the computer models, the engineering of concepts to the required detail, and the use of a verification analysis loop.

## TABLE OF CONTENTS

### Naval Engagement Model (NEM)

### Appendices

Volume IIB (Unclassified)
Appendices

Volume IIC (Secret)

Volume IID (Unclassified)

TITLE

NAVAL ENGAGEMENT MODEL (NEM)
FORTRAN SOURCE PROGRAM

Appendix N

NO.

DATE January 1974

## TABLE OF CONTENTS

Appendix N completely revised.

PREPARED BY R. C. Dyer

APPROVED BY L. D. Gregory

PAGE N-1 OF N-202

(Reverse side blank)

# APPENDIX N

# NAVAL ENGAGEMENT MODEL (NEM)
# FORTRAN SOURCE PROGRAM

## 1. INTRODUCTION

Appendix N gives the source program listing for the Naval Engagement Model (NEM). The NEM was written in Fortran IV EBCDIC for the IBM 360 computer. Approximately 9500 source cards make up the 81 subprograms of the NEM.

The listing in Section 2.0 reflects the status of the NEM as delivered to DIA on 19 Jan 1974. The NEM subprograms are grouped by function (related routines are grouped together) and listed in the approximate order of execution. Section 3.0 provides an index to the listing by function. Section 4.0 provides an alphabetical index.

(Reverse side blank)

2. LISTING

```
C   PGM=NXX(NFM)       MAIN PROGRAM
C   MODS.MAIN,KILLHF,STATE,RADAR,FFACT,SEAREF.               VER.13   11-25-74
C   ADD .STRING,SIGOS.                                        VER.13   11-25-74
C
C         DEFINE FILE 4(1000,40,U,JD4)
          DEFINE  FILE 11( 180,555,U,JD11)
        COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12
          INTEGER  ZIP,ZCODE,Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9
          COMMON/INOUT/NLINE,NPAGE,PCODE(20),MISC(7),XMISC(7),ZIP,ZCODE(19)
       1, JRASH(20),TRASH(20),IR(8),IC(8),DUM(8),IDUM(8),NFLAG,NFLAG2
          EQUIVALENCE  (ZCODE(1),Z1),(ZCODE(2),Z2),(ZCODE(3),Z3)
       1                ,(ZCODE(4),Z4),(ZCODE(5),Z5),(ZCODE(6),Z6)
       2                ,(ZCODE(7),Z7),(ZCODE(8),Z8),(ZCODE(9),Z9)
          COMMON/KLAS/ CLAS(20)
          COMMON/INOUU/IPR(16),JPAR(16),PAR(16),LABEL
          COMMON/INOU4/JD4
C         COMMON/INOU9/JDS,ND9,NB9,INDX9(   10,6)
          COMMON/INOU11/ JD11,ND11,NB11,INDX11(180,7)
          NAMELIST/NAM1/MISC,XMISC,IPR,JPAR,PAR,NPAGE,LABEL
CFORMATS
 1000   FORMAT(1X,19A4,A3)
 2000   FORMAT(1H1,71X, 4HPAGE,I4/6X,19A4,A3)
 1002   FORMAT(A4,3I2,I10,4I5,10A4)
 2002   FORMAT(/6X,A4,3I2,I10,4I5,10A4)
 1003   FORMAT (10X,7G10.3)
 2003   FORMAT (10X,7G13.6)
 2005   FORMAT(1H0,20HERROR IN MAIN AT E1=,F8.2,3X,A4,3I2,I10,4I5,10A4)
 2006   FORMAT(//////////44X, 4HNXX //
     1         44X, 4HXXXX//
     2         35X,'VOUGHT SYSTEMS DIVISION',  //
     3         34X,25HLTV AEROSPACE CORPORATION            //
     4         37X,19HDALLAS, TEXAS 75222 )
C***  SAMPLE ZIPS FOR BASIC TABLE HANDLING
CIP  3                            START NEW BASIC TABLE FILE(REPLACE OLD)
CIP  7 1 1                        READ BASIC TABLE & ADD TO DISK 11
CIP  7 1 2                        FIND AND PRINT FOLLOWING BASIC TABLE
CABLE NO.   ID1.....     ID2        (ONE ID CARD)
CIP  7 1 4                     -1 UPDATE BASIC TABLE.TEMP IF -1 IN COL.30
CIP  7 1 5                        REPLACE FOLLOWING BASIC TABLE ON DISK 11
CIP  7 1 6                        PRINT ALL BASIC TABLES NOW ON DISK11
CIP  7 3                          DELETE ALL TEMPORARY TABLES ON DISK11
CIP 11 1 P  ID1.....   -N         UPDATE ORIG. TABLE=ID1,EXT=-N,NAMELIST
CIP 11 2 P  ID1.....   -N         UPDATE BASIC TABLE UPDATE IF ANY
C***
          DATA  BLANK/'    '/
          NFLAG = 0
          NFLAG2= 0
          ND9   = 1
          NB9   = 0
          LAA1 =  27
          MAA1 =  12
          KZARL=  96 + LAA1*(5+MAA1)
          IMAGE = 1
          JD11 = 1
          READ (N11'JD11,ERR=1)  ND11,NB11,INDX11
 1    NPAGE = 1
 101  READ (N5,1000) CLAS, PCODE
      IF (PCODE(1).EQ.BLANK.OR.IMAGE.EQ.0)  IMAGE=0
 2    CALL PAGE
      WRITE(N6,2006)
 900  CALL PAGE
      IF ( IMAGE.EQ.1)  GO TO 38
 902  F1 = 902.
      READ(N5,1002,END= 10,ERR=9021) ZCODE
      IF ( IPR(1).EQ.0)  GO TO 9022
      WRITE(N6,2002) ZCODE
      NLINE = NLINE + 2
C**
 9022 GO TO (1,2,3,4,5,6,7,8,9,10,11,12) ,Z2
 9021 NFLAG = NFLAG + 1
      WRITE(N6,2005) F1, ZCODE
      GO TO 902
C**READ INOUT PARAMETERS
 3    ND11 = 3
      NB11 = 0
      GO TO 902
 38   CALL  CARD(N5,N6,N9, CLAS,PCODE)
      IMAGE = 0
      N5  = N9
      GO TO 101
 4    IF (Z3.GT.1)  GO TO 42
      READ (N5,NAM1)
      WRITE(N6,NAM1)
      GO TO 900
 42   GO TO 902
 5    GO TO 902
 6    CALL  LCHEK( ZCODE ,N5,N6)
      GO TO 902
C**  READ  ASIC TABLES AND STORE ON DA DISK 11
 7    IX = Z4
      IF (Z3.GE.2)  GO TO 73
      IPR(1) = 0
      IF (Z4.NE.1)   IPR(1)=1
      CALL  TBASIC(IX)
      GO TO 902
C***  DELETE ALL TEMPORARY TABLES ON DISK11
 73   K   = 0
      DO 7301 I=1,NB11
      IF ( INDX11(I,7).GE.0)  GO TO 7301
      K   = K + 1
      INDX11(I,2) = 0
 7301 CONTINUE
      NB11 = NB11 - K
      IF (K.GT.0)  ND11=INDX11(NB11+1,4)
      IF (Z2.EQ.10)  GO TO 1001
      GO TO 902
 8    IX  = Z4
      IPR(1) = 1
      IF (Z3.LE.0)  GO TO 8C
      GO TO (81,82,83,84,85,86), Z3
```

```
C                                                                       NXX 1110
 80     CALL   PAGE                                                     NXX 1120
        CALL   SETUPA(IX)                                               NXX 1130
        CALL   SETUPB(IX)                                               NXX 1140
        CALL   SETUPC(IX)                                               NXX 1150
        CALL   SETUPD(IX)                                               NXX 1160
        CALL  SETUPF(IX)                                                NXX 1170
       CALL   SETUPE(IX)                                                NXX 1180
        CALL   ALLXYZ(-4.,1,N6,IXX)                                     NXX 1190
        CALL  ALLXYZ( 0.,1,N6,IXX)                                      NXX 1200
        GO TO 902                                                       NXX 1210
 81     IF (NLINE.GT.10)  CALL PAGE                                     NXX 1220
        CALL SETUPA(IX)                                                 NXX 1230
      CALL   ALLXYZ(.0,1,N6,IXX)                                        NXX 1240
        CALL   ALLXYZ(-.25,1,N6,IXX)                                    NXX 1250
        GO TO 902                                                       NXX 1260
 82     CALL   SETUPB(IX)                                               NXX 1270
        GO TO 902                                                       NXX 1280
 83     CALL SETUPC(IX)                                                 NXX 1290
        GO TO 902                                                       NXX 1300
 84     CALL   SETUPD(IX)                                               NXX 1310
        GO TO 902                                                       NXX 1320
   85   CALL SETUPF(IX)                                                 NXX 1330
        GO TO 902                                                       NXX 1340
 86     CALL   SETUPE(IX)                                               NXX 1350
        GO TO 902                                                       NXX 1360
 9      CALL   XECUTE                                                   NXX 1370
        GO TO 902                                                       NXX 1380
 10     GO TO 73                                                        NXX 1390
 1001   JD11 = 1                                                        NXX 1400
        WRITE(N11'JD11)  ND11,NB11,INDX11                               NXX 1410
        WRITE(N6,2003)  ND9,NB9,ND11,NB11                               NXX 1420
        STOP                                                            NXX 1430
 11     CALL   UPDA(Z3,Z4,Z5,Z6,IXX)                                    NXX 1440
        IF (IXX.GT.0)   NFLAG=NFLAG+1                                   NXX 1450
        GO TO 902                                                       NXX 1460
 12     IF (Z4.GE.0)   GO TO 1201                                       NXX 1470
        N5   = NN5                                                      NXX 1480
        REWIND  N8                                                      NXX 1490
        GO TO 902                                                       NXX 1500
 1201   NN5 = N5                                                        NXX 1510
        CALL   STRING(Z3,Z4,Z5,Z6,N5,N6,N8,ZCODE)                       NXX 1520
        GO TO 902                                                       NXX 1530
        END                                                             NXX 1540


        SUBROUTINE PAGE                                                 PAGE0010
C** PGM=NXX          L.D.GREGORY. VER.1  11-15-72. FORTRAN IV. . EBCC    PAGE0020
        COMMON/INOUT/NLINE,NPAGE,PCODE(20),IDUM(108)                    PAGE0030
        COMMON/KLAS/ CLAS(20)                                           PAGE0040
 1000   FORMAT(//26X,19A4,A3/1H1,25X,19A4,A3/)                          PAGE0050
 2000   FORMAT(6X,19A4,A3,6X, 4HPAGE,I4/)                               PAGE0060
        WRITE( 6,1000) CLAS, CLAS                                       PAGE0070
        WRITE( 6,2000) PCODE,NPAGE                                      PAGE0080
      NPAGE = NPAGE + 1                                                 PAGE0090
      NLINE = 8                                                         PAGE0100
      RETURN                                                            PAGE0110
      END                                                               PAGE0120


      SUBROUTINE WCBD11(BZA,LZA,MZA,NCB,KZRL,ID1,ID2,ID3,IX,IXX)        WCBD0010
     COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              WCBD0020
C   PGM=NXX(NEM) MODS.INDX11(180)=300.                  VER.3    1-10-75 WCBD0030
C   PGM=NU5.  SEM.                         VER.2. 04-22-71.FORTRAN IV.FBCD.WCBD0040
C   UTILITY 360 O/S. VER.2. 02-19-71.     L.D.GREGORY,UNIT 3-53300,EXT 510WCBD0050
C   BZA IS BZA(LZA,MZA)                                                 WCBD0060
C   TO WRITE COMMON BLOCKS ON DIRECT ACCESS(DISK) FILE 11               WCBD0070
C   NCB  = COMMON BLK NO., =1 FOR COMMON/CZAZA/, =2 FOR /CZAZB/, ETC.   WCBD0080
C   ID1  = IDENTIFIER NO.1 (PERHAPS AN INDENTURED CODE) OF THIS LOAD    WCBD0090
C   ID2  = IDENTIFIER NO.2 (PERHAPS THE SEQUENCE NO.  ) OF THIS LOAD    WCBD0100
C   ID3  = IDENTIFIER NO.3 (PERHAPS THE DATE OF THE DATA)               WCBD0110
C        =  OR, USE AS FIRST AND LAST NOS. OF TABLES IN THIS LOAD       WCBD0120
C   IX   = ON INPUT(=1 USE ID1,=2 USE ID2,=3 USE BOTH,=4 USE RANGE)     WCBD0130
C   IXX  = ON OUTPUT(=0 LOADED O.K.,=1 TROUBLE)                         WCBD0140
C   KZRL = NO.OF 4-BYTE WORDS IN EACH LOAD                              WCBD0150
C   JD11  = FILE 11 ASSOC.VARIABLE. SEE MAIN AND   //GO.FT11F001  DD    WCBD0160
C* ND11  = VALUE OF FIRST FREE RECORD ON DISK 11                        WCBD0170
C   NB11  = CURRENT(AND FINAL) NO. OF LOADS ON FILE 11                  WCBD0180
C   LZ11  = NO OF 4-BYTE WORDS IN FIXED LENGTH RECORD ON FILE 11        WCBD0190
C   MZ    = NO OF LOGICAL RECORDS EACH OF LENGTH LZ11 IN LOAD'I'.       WCBD0200
C   KEEP  LZA = LZ11                                                    WCBD0210
C   INDX11(I,J) ,     FOR I = LOAD NO.(EACH TIME A COMMON BLK IS EMPTIED WCBD0220
C                           = NB11(CURRENT)            IS ONE LOAD).    WCBD0230
C     FOR  J=1  , =  NCB, COMMON BLK NO.                                WCBD0240
C          J=2  , =  ID1, IDENTIFIER NO. 1                              WCBD0250
C          J=3  , =  ID2, IDENTIFIER NO. 2                              WCBD0260
C          J=4  , =  ND11  =  START OF TABLE NB11 ON DISK 11            WCBD0270
C          J=5  , =  KZRL  =  SEE ABOVE                                 WCBD0280
C          J=6  , =  BLKSIZE ON DISK 11                                 WCBD0290
C          J=7  , =  ID3, IDENTIFIES NO. 3                              WCBD0300
      COMMON/INOU11/ JD11,ND11,NB11,INDX11(180,7)                       WCBD0310
      DIMENSION  BZA(KZRL)                                              WCBD0320
      DATA  LINDX/ 180/                                                 WCBD0330
 2000 FORMAT(1H0,'BASIC TABLE INDEX=INDX11(    ,7) IS FULL AT NO. OF TABWCBD0340
     1LES=', I4/1X,'ID1, ID2=', 2I12)                                   WCBD0350
C***                                                                    WCBD0360
C***                                                                    WCBD0370
 1    NB11 =  NB11 + 1                                                  WCBD0380
      IF (NB11.GT.LINDX)   GO TO 12                                     WCBD0390
      INDX11(NB11,1) = NCB                                              WCBD0400
      INDX11(NB11,2) = ID1                                              WCBD0410
      INDX11(NB11,3) = ID2                                              WCBD0420
      INDX11(NB11,4) = ND11                                             WCBD0430
      MZ             = (KZRL-1)/LZA   + 1                               WCBD0440
      INDX11(NB11,5) = KZRL                                             WCBD0450
      INDX11(NB11,6) = LZA                                              WCBD0460
      INDX11(NB11,7) = ID3                                              WCBD0470
C***  UNLOAD  BZA                                                       WCBD0480
```

```
  8    JD11 = ND11                                                      WCBD0490
 10    WRITE(N11'JD11) BZA                                              WCBD0500
       ND11 = JD11                                                      WCBD0510
       IXX= 0                                                           WCBD0520
       RETURN                                                           WCBD0530
 12    WRITE(N6,2000)  NB11,ID1,ID2                                     WCBD0540
       STOP                                                             WCBD0550
C***                   ***                     ***                      WCBD0560
C***                   ***                     ***                      WCBD0570
       ENTRY       RCBD11(BZA,LZA,MZA,NCB,KZRL,ID1,ID2,IX,IXX)          WCBD0580
C                                                                       WCBD0590
C  PGM=NU5.  SEM.                            VER.2. 04-22-71.FORTRAN IV.EBCD.WCBD0600
C UTILITY 360.VER.2.      02-19-71  L.D.GREGORY     FORTRAN IV      EBCD  WCBD0610
C  TO READ FROM DIRECT ACCESS(DISK)  FILE 11 INTO COMMON BLOCK NCB=1,2  WCBD0620
C  NCB, ID1, ID2, IX - SAME AS ABOVE.         KZRL= DUMMY               WCBD0630
C  MATCH ID CODES                                                       WCBD0640
       KRW = 0                                                          WCBD0650
 29    ASSIGN  46 TO KS1                                                WCBD0660
       IF (IX.EQ.3)  ASSIGN  32 TO KS1                                  WCBD0670
       GO TO (30,40,30), IX                                             WCBD0680
 30    DO 34 J=1,NB11                                                   WCBD0690
       I = NB11 - J + 1                                                 WCBD0700
       IF (ID1.NE.INDX11(I,2) )  GO TO 34                               WCBD0710
       GO TO KS1, (32,46)                                               WCBD0720
 32    IF (ID2.NE.INDX11(I,3) )  GO TO 34                               WCBD0730
       GO TO 46                                                         WCBD0740
 34    CONTINUE                                                         WCBD0750
       IF (KRW.EQ.1)   GO TO 1                                          WCBD0760
       GO TO 43                                                         WCBD0770
 40    DO 42 J=1,NB11                                                   WCBD0780
       I = NB11 - J + 1                                                 WCBD0790
       IF (ID2.EQ.INDX11(I,3) )  GO TO 46                               WCBD0800
 42    CONTINUE                                                         WCBD0810
       IF(KRW.EQ.1) GO TO 1                                             WCBD0820
 43    IXX= 1                                                           WCBD0830
       RETURN                                                           WCBD0840
C***   ID CODES MATCH                                                   WCBD0850
 46    JB11 = I                                                         WCBD0860
       KZ   = INDX11(JB11,5)                                            WCBD0870
     JD11 = INDX11(JB11,4)                                              WCBD0880
       IF (KRW.NE.0)  GO TO 60                                          WCBD0890
C***   LOAD IN BZA                                                      WCBD0900
 52    READ(N11'JD11) BZA                                               WCBD0910
       IXX= 0                                                           WCBD0920
       RETURN                                                           WCBD0930
C***                   ***                     ***                      WCBD0940
C***                   ***                     ***                      WCBD0950
       ENTRY       WWBD11(BZA,LZA,MZA,NCB,KZRL,ID1,ID2,ID3,IX,IXX)      WCBD0960
C                                                                       WCBD0970
C  PGM=NU5.  SEM.                            VER.2. 04-22-71.FORTRAN IV.EBCD.WCBD0980
C UTILITY 360.VER.2.      02-19-71  L.D.GREGORY     FORTRAN IV      EBCD  WCBD0990
C  TO RE-WRITE ONTO DIRECT ACCESS(DISK)  FILE 11 INTO SAME SPACE        WCBD1000
C  MATCH ID CODES                                                       WCBD1010
       KRW = 1                                                          WCBD1020
       GO TO 29                                                         WCBD1030
```

```
C***  LOAD IN BZA                                                       WCBD1040
 60   WRITE(N11'JD11) BZA                                               WCBD1050
      IXX= 0                                                            WCBD1060
      INDX11(JB11,7) = ID3                                              WCBD1070
      RETURN                                                            WCBD1080
      END                                                               WCBD1090


      SUBROUTINE  WMAT3(KIND,IX,X,LA,MA,NA,L,M,N,NLINE,TTL)             WMAT0010
C   PGM=NXX  VER.3.  4-25-73  NEM  L.D.GREGORY  IBM 360   EBCD          WMAT0020
C   UTILITY TRIPLEX. VER.1.  7/19/70.  FORTRAN IV.      EBCD            WMAT0030
C   PURPOSE. WRITE A MATRIX  N  FROM A SERIES OF  NA  SUCH.             WMAT0040
C   KIND = 1, INTEGER                                                   WMAT0050
C        = 2, REAL                                                      WMAT0060
      DIMENSION  IX(LA,MA,NA), X(LA,MA,NA), TTL(15)                     WMAT0070
 2000 FORMAT(1H1,71X, 4HPAGE,I4/6X,19A4,A3)                             WMAT0080
 2002 FORMAT(/6X,12HMATRIX NO.= ,I4/6X,15A4)                            WMAT0090
 2004 FORMAT(6X,4H ROW,10(2X,4HCOL.,I2,4X) )                            WMAT0100
 2007 FORMAT(5X,I5,10G12.5/(10X,10G12.5) )                              WMAT0110
 2008 FORMAT(5X,I3,I10,9I12/(6X,10I12))                                 WMAT0120
      ASSIGN 141 TO KS1                                                 WMAT0130
      IF (KIND.EQ.2)  ASSIGN 142 TO KS1                                 WMAT0140
      INCR = M/10 + 1                                                   WMAT0150
      INCM = INCR*L + 6                                                 WMAT0160
      IF (NLINE + INCM - 61)  6, 6, 4                                   WMAT0170
 4    CALL PAGE                                                         WMAT0180
C                                        DELETED                        WMAT0190
      NLINE = 9                                                         WMAT0200
 6    NLINE = NLINE + 4                                                 WMAT0210
      WRITE(6,2002)   N,TTL                                             WMAT0220
      K    = MIN0(10,M) - 1                                             WMAT0230
      WRITE(6,2004) (J,J=1,K)                                           WMAT0240
      I    = 0                                                          WMAT0250
 14   I    = I + 1                                                      WMAT0260
      GO TO KS1, (141,142)                                              WMAT0270
 141  WRITE(6,2008) I,(IX(I,J,N),J=1,M)                                 WMAT0280
      GO TO 143                                                         WMAT0290
 142  WRITE(6,2007) I,( X(I,J,N),J=1,M)                                 WMAT0300
 143  NLINE = NLINE + INCR                                              WMAT0310
      IF (I-L) 16,18,18                                                 WMAT0320
 16   IF (NLINE + INCR - 60) 14,14, 4                                   WMAT0330
 18   RETURN                                                            WMAT0340
      END                                                               WMAT0350


      SUBROUTINE  WRITRX(X,LXA,MXA,L,M,VAR,INX,NLINE,NPAGE,PCODE,TITLE)WRRX0010
C   LA0268. VER. 2  08/18/70 SCREENING.  L.D.GREGORY 3-55100 EXT. 510   WRRX0020
C   UTILITY 360/65. SUBR.WRITRX.VER.1. 12/06/69. FORTRAN IV       EBCD  WRRX0030
      DIMENSION  X(LXA,MXA),TITLE(15),PCODE(20)                         WRRX0040
 2000 FORMAT(1H1,71X, 4HPAGE,I4/6X,19A4,A3)                             WRRX0050
 2002 FORMAT(/6X,4HVAR=,F12.5/6X,15A4)                                  WRRX0060
 2004 FORMAT(6X,4H ROW,10(2X,4HCOL.,I2,4X) )                            WRRX0070
 2007 FORMAT(5X,I5,10G12.5/(10X,10G12.5) )                              WRRX0080
```

```
      I  = 0                                                            WRRX0090
 1    INX = 0                                                           WRRX0100
      INCR = M/10 + 1                                                   WRRX0110
      INCM = INCR*L + 6                                                 WRRX0120
      IF (NLINE + INCM - 61)  6, 6, 4                                   WRRX0130
 4    CALL PAGE                                                         WRRX0140
      NLINE = 8                                                         WRRX0150
 6    NLINE = NLINE + 4                                                 WRRX0160
      WRITE(6,2002) VAR,TITLE                                           WRRX0170
      K    = MINO(10,M) - 1                                             WRRX0180
      WRITE(6,2004) (J,J=1,K)                                           WRRX0190
 14   I  = I + 1                                                        WRRX0200
      WRITE(6,2007) I,(X(I,J),J=1,M)                                    WRRX0210
      NLINE = NLINE + INCR                                              WRRX0220
      IF ( I-L) 16,18,18                                                WRRX0230
 16   IF (NLINE + INCR - 60) 14,14, 4                                   WRRX0240
 18   RETURN                                                            WRRX0250
      END                                                               WRRX0260


      SUBROUTINE  SORT(VALUE,IRANK,ITEM,N)                              SORT0010
C  LA0268. VER. 2  08/18/70 SCREENING.  L.D.GREGORY 3-55100 EXT. 510    SORT0020
C  UTILITY 360/65 OR RAX.  1/05/70  FIXED DIM. FORTRAN BASIC.  EBCD     SORT0030
C       PURPOSE.  TO TAKE N  ITEMS EACH WITH VALUE(I), I=1,N            SORT0040
C                 AND RANK IN ORER FROM SMALLEST TO LARGEST             SORT0050
C                 AND STORE RANK OF THE  ITH. ITEM IN  IRANK(I)         SORT0060
C                 AND STORE NO. OF ITEM WITH  JTH. RANK IN  ITEM(J).    SORT0070
     DIMENSION VALUE(300), IRANK(300), ITEM(300)                        SORT0080
      IF(N-1) 52,52,1                                                   SORT0090
 1    DO 10 I=1,N                                                       SORT0100
 10   IRANK(I) = 1                                                      SORT0110
 14   N1 = N-1                                                          SORT0120
      DO  40  I = 1,N1                                                  SORT0130
      J1 = I + 1                                                        SORT0140
      DO  40  J = J1,N                                                  SORT0150
 16   IF (VALUE(I)-VALUE(J)) 30,30,20                                   SORT0160
 20   IRANK(I) = IRANK(I) + 1                                           SORT0170
      GO TO 40.                                                         SORT0180
 30   IRANK(J) = IRANK(J) + 1                                           SORT0190
 40   CONTINUE                                                          SORT0200
      DO  50  I = 1,N                                                   SORT0210
      J  = IRANK(I)                                                     SORT0220
 50   ITEM(J) = I                                                       SORT0230
      RETURN                                                            SORT0240
 52   ITEM(1) = 1                                                       SORT0250
      IRANK(1)= 1                                                       SORT0260
      RETURN                                                            SORT0270
      END                                                               SORT0280


      SUBROUTINE  KORDER(KA,NA)                                         ORDR0010
C  PGM=NXX.  L.D.GREGORY. VER.1   3-8-73.   FORTRAN IV.    EBCD         ORDR0020
C  TO ORDER FROM SMALLEST TO LARGEST  -  INTEGERS                       ORDR0030
```

```
      DIMENSION  KA(NA)                                                 ORDR0040
 1    IF (NA.LE.1)  RETURN                                              ORDR0050
      NA1 = NA - 1                                                      ORDR0060
      DO 10 I=1,NA1                                                     ORDR0070
      L   = KA(I)                                                       ORDR0080
      JA  = I + 1                                                       ORDR0090
      DO 10 J=JA,NA                                                     ORDR0100
      IF (L .LE.KA(J))  GO TO 10                                        ORDR0110
      KA(I) = KA(J)                                                     ORDR0120
      KA(J) = L                                                         ORDR0130
      L   = KA(I)                                                       ORDR0140
 10   CONTINUE                                                          ORDR0150
      RETURN                                                            ORDR0160
      ENTRY  RORDER(XA,NA)                                              ORDR0170
C  TO ORDER FROM SMALLEST TO LARGEST  -  REALS                          ORDR0180
      DIMENSION  XA(NA)                                                 ORDR0190
 21   IF (NA.LE.1)  RETURN                                              ORDR0200
      NA1 = NA - 1                                                      ORDR0210
      DO 28 I=1,NA1                                                     ORDR0220
      X   = XA(I)                                                       ORDR0230
      JA  = I + 1                                                       ORDR0240
      DO 28 J=JA,NA                                                     ORDR0250
      IF (X .LE.XA(J))  GO TO 28                                        ORDR0260
      XA(I) = XA(J)                                                     ORDR0270
      XA(J) = X                                                         ORDR0280
      X   = XA(I)                                                       ORDR0290
 28   CONTINUE                                                          ORDR0300
      RETURN                                                            ORDR0310
      END                                                               ORDR0320


      SUBROUTINE  TBASIC(IX)                                            TBAS0010
C  PGM=NXX.          L.D.GREGORY    VER.3.  7-20-73  FORTRAN IV. EBCD   TBAS0020
C MODS. INDX11(180)=250.                          VER.4    1-10-75 TBAS0030
C  TO READ INDENTURED CODED TABLES AND STORE ON DA DISK 11.             TBAS0040
CIP  7 1 1                               READ ONTO DISK 11              TBAS0050
CIP  7 1 2                                                              TBAS0060
CIP  7 1 3                                                              TBAS0070
CIP  7 1 4                 -1           TABLE UPDATE                    TBAS0080
CIP  7 1 5                               REPLACE TABLE                  TBAS0090
CIP  7 1 6                               PRINT ALL TABLES NOW ON DISK 11TBAS0100
C******************************                     ***************     TBAS0110
C  PGM=NU5(SEM) STRATEGIC ENGAGEMENT MODEL - 1 JUNE 71 FORTRAN IV EBCD  TBAS0120
C  EXECUTIVE SUBPROGRAM FOR THE BASIC TABLE PROCESSING LINK             TBAS0130
     COMMON/BASTAB/ KZARL                                               TBAS0140
C  PGM=NU5. SEM                          VER.2. 04-05-71.FORTRAN IV.EBCD.TBAS0150
C  L.D.GREGORY,UNIT 3-53300,EXT 510.                                    TBAS0160
C  SUBROUTINE PURPOSE, TO READ A SET OF BASIC LABELED TABLES.           TBAS0170
      INTEGER  ZIP,ZCODE,Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9                     TBAS0180
     COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              TBAS0190
      COMMON/INOUT/NLINE,NPAGE,PCODE(20),MISC(7),XMISC(7),ZIP,ZCODE(19)TBAS0200
    1, JRASH(20),TRASH(20),IR(8),IC(8),DUM(8),IDUM(8),NFLAG,NFLAG2      TBAS0210
      EQUIVALENCE  (ZCODE(1),Z1),(ZCODE(2),Z2),(ZCODE(3),Z3)            TBAS0220
    1              ,(ZCODE(4),Z4),(ZCODE(5),Z5),(ZCODE(6),Z6)           TBAS0230
```

```
     2                  ,(ZCODE(7),Z7),(ZCODE(8),Z8),(ZCODE(9),Z9)
      COMMON /INOUU/ IPR(16),JPAR(16), PAR(16),LABEL
       COMMON/INOU11/ JD11,ND11,NB11,INDX11(180,7)
CZAZA
       INTEGER  TITLA
       COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)
     1, TITLA(15),HEADA(37),IFMT
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)
       DIMENSION  BZA( 555)
       EQUIVALENCE  (NCBA,BZA(1))
      DIMENSION LABLA(4,8),LABLB(52)
      EQUIVALENCE(LABLA(1),ICA(1)),(LABLB(1),TITLA(1))
      DATA NBLANK/'    '/
C  NOTE. IF BZA(LZA,MZA), THEN LZA=RECORD LENGTH ON FILE 11,
C  AND MZA=ENOUGH TO COVER UP TO PADZA BUT NOT MORE THAN ALL OF CZAZA.
C  LA1,LAA1,MA1,MAA1 ARE ACTUAL AND ABSOLUTE DIMENSIONS OF VA1,
C   AND LA1,LAA1 ARE ACTUAL AND ABSOLUTE LENGTH OF KA1.
C  KZARL=NO.OF 4 BYTE WORDS IN CZAZA UP TO PADZA
C  IDA1 AND IDA2 ARE INPUT NUMBERS TO UNIQUELY IDENTIFY THE TABLE=LOAD
C   IN CZAZA AT ANY GIVEN TIME.
C
C  PRINT CONTROL,   IPR(2) = 0, NO PRINT UNLESS(Z4.NE.1)
C                          = 1, PRINT
C  IF Z4 = 1, READ FROM FILE 5 AND USE IPR(2) PRINT OPTION
C          2, WRITE ONLY (FILE 6) = PRINT
C          3, READ FROM FILE 5 AND PRINT
C  IF Z4 = 4, READ FROM FILE 5 AND UPDATE, TEMP IF Z7.LT.0
C        = 5, REPLACE ON DISK 11, ADD IF NEW
C        = 6, PRINT ALL BASIC TABLES (INPUTS) NOW STORED ON DISK 11
C  IF IPR(3) = 0, USE STD. COL HEADS IN PRINT (IF PRINTED)
C            = 1, USE HEADS READ IN
CFORMATS
 1000  FORMAT(12X,2I8,I2,2I5,10A4)
 1002  FORMAT(5A4,6(2X,2A4))
 1003  FORMAT((20X,6(2X,2A4)) )
 1004  FORMAT(I3,1X,3A4,I4,6F10.0)
 1005  FORMAT((20X,6F10.0))
 1006  FORMAT(4(2A4,A2,I10))
 1007  FORMAT(4(2A4,A2,F10.0))
 2005  FORMAT(1H0,22HERROR IN BASIC  AT E1=F8.2,3X,A4,3I2,I10,4I5,10A4/
     1         ( 6X,3HE 2=10G12.5) )
C** INITIALIZE. NOTE,IF BZA(LZA,MZA), THEN LENGTH OF DATA IN CZAZA
C** IS KZARL WHICH IS.LE.(LZA*MZA).  VA1 IS VA1(LAA1,MAA1).
 1     NCBA = 1
       LZA  = 555
       MZA  = 12
       LAA1 = 27
       MAA1 = 12
      KZARL=96+5*LAA1+LAA1*MAA1
       JZARL = KZARL
       NFLAG2 = 0
       IF (Z4.EQ.6)  GO TO  150
C** READ TABLES AND STORE
       READ(N5,1000) TITLA
       IDA1 = TITLA(1)
      IDA2 = TITLA(2)
      IDA3 = TITLA(3)
      IDA4 = TITLA(4)
      MA1  = TITLA(5)
      Z9   = MA1
      GO TO (4,100,4,120,4), Z4
 4    K3 = 2 * Z9 + 5
      K4 = MIN0(Z9, 6)
      ASSIGN 14 TO KS1
      IF (Z9.GT.6)  GO TO 7
      ASSIGN  10 TO KS1
 7    READ(N5,1002) (HEADA(I),I=1,17)
      IF (Z9.LE.6)  GO TO 9
      READ(N5,1003) (HEADA(I),I=18,K3)
 9    IF (Z4.EQ.4)  GO TO 124
      I = 0
 10   I = I + 1
      READ(N5,1004) (KA1(I,J),J=1,5)    , (VA1(I,J),J=1,K4)
      IF (KA1(I,1).LE.0)  GO TO 16
      GO TO KS1, (14,10)
 14   READ(N5,1005) (VA1(I,J),J=7,Z9)
      GO TO 10
C** FILL LABELS IN CZAZA
 16   LA1 = I - 1
 18   READ(N5,1006) ICA
      READ(N5,1007)  CA
     IF(LABEL.GT.0) GO TO 185
     DO 181 II=1,8
     DO 181 JJ=1,3
  181 LABLA(JJ,II)=NBLANK
     DO 182 II=6,52
  182 LABLB(II)=NBLANK
     DO 183 II=2,4
     DO 183 JJ=1,LA1
  183 KA1(JJ,II)=NBLANK
 185  IY = 3
      IF (Z4.EQ.3) GO TO 102
      IF (Z4.EQ.1) GO TO 28
      IF (Z4.EQ.4 .AND. Z7.LT.0) GO TO 28
      IF (Z4.EQ.4 .AND. Z7.GE.0) GO TO 26
      IF (Z4.EQ.5) GO TO 26
   25 E1=25.
      GO TO 31
C** LOAD ON DISK 11
 26   IY = 3
      CALL  WWBD11(BZA,LZA,MZA,NCBA,KZARL,IDA1,IDA2,IDA3,IY,IXX)
      IF(IXX.EQ.0) GO TO 32
      GO TO 30
   28 CALL  WCBD11(BZA,LZA,MZA,NCBA,KZARL,IDA1,IDA2,IDA3,IY,IXX)
      IF (IXX.EQ.0)  GO TO 32
 30   E1 = 30.
 31   WRITE(N6,2005)E1,ZCODE,IDA1,IDA2
      NFLAG = NFLAG  + 1
      RETURN
C**  TEST FOR PRINT
```

```
 32     IF (IPR(2).EQ.0.AND.Z4.LE.1)  RETURN                                         TBAS1340
        CONTINUE                                                                     TBAS1350
        IF (IPR(2).LE.1.OR.IPR(2).GE.3)  GO TO 102                                   TBAS1360
C**  CALL FROM DISK 11 AND PRINT OUT                                                 TBAS1370
 100    IY = 3                                                                       TBAS1380
        CALL   RCBD11(BZA,LZA,MZA,NCBA,KZARL,IDA1,IDA2,IY,IXX)                       TBAS1390
        IF (IXX.EQ.0)  GO TO 102                                                     TBAS1400
        F1 = 100.03                                                                  TBAS1410
        GO TO 31                                                                     TBAS1420
 102    IY = IPR(3)                                                                  TBAS1430
        CALL   WBASIC(KA1,VA1,LAA1,MAA1,LA1,MA1,IDA1,IDA2,IDA3,ICA,CA,IY             TBAS1440
     1,         TITLA,HEADA )                                                        TBAS1450
        IF (IPR(2).LE.2)  CALL PAGE                                                  TBAS1460
        RETURN                                                                       TBAS1470
C**  RECALL FROM FILE 11 AND UPDATE                                                  TBAS1480
 120    IY = 3                                                                       TBAS1490
        CALL   RCBD11(BZA,LZA,MZA,NCBA,KZARL,IDA1,IDA2,IY,IXX)                       TBAS1500
        IF (IXX.EQ.0)  GO TO 122                                                     TBAS1510
C                                                                                    TBAS1520
        F1 = 120.04                                                                  TBAS1530
        GO TO 31                                                                     TBAS1540
 122    IDA3 = Z7                                                                    TBAS1550
        GO TO 4                                                                      TBAS1560
 124    READ(N5,1004) (IR(J),J=1,5) , (TRASH(J),J=1,6)                               TBAS1570
        IF (IR(1).LE.0)  GO TO 18                                                    TBAS1580
        IF (Z9.LE.6)  GO TO 125                                                      TBAS1590
        READ(N5,1005) (TRASH(J),J=7,Z9)                                              TBAS1600
 125    DO 126 I=1,LA1                                                               TBAS1610
        IF (KA1(I,1).EQ.IR(1))  GO TO 128                                            TBAS1620
 126    CONTINUE                                                                     TBAS1630
C**  ADD AT END OF TABLE                                                             TBAS1640
        LA1 = LA1 + 1                                                                TBAS1650
        I   = LA1                                                                    TBAS1660
 128    DO 130 J=1,5                                                                 TBAS1670
 130    KA1(I,J) = IR(J)                                                             TBAS1680
        DO 132 J=1,Z9                                                                TBAS1690
 132    VA1(I,J) = TRASH(J)                                                          TBAS1700
        GO TO 124                                                                    TBAS1710
C**  PRINT ALL NOW ON DISK 11                                                        TBAS1720
 150    CALL PAGE                                                                    TBAS1730
        DO 152 I=1,NB11                                                              TBAS1740
        IF (INDX11(I,1).NE.NCBA) GO TO 152                                           TBAS1750
        IF (INDX11(I,2).LE. 0)  GO TO 152                                            TBAS1760
        JD11=INDX11(I,4)                                                             TBAS1770
        READ(N11'JD11)  BZA                                                          TBAS1780
        IY=IPR(3)                                                                    TBAS1790
        CALL   WBASIC(KA1,VA1,LAA1,MAA1,LA1,MA1,IDA1,IDA2,IDA3,ICA,CA,IY             TBAS1800
     1,        TITLA,HEADA)                                                          TBAS1810
C       CALL PAGE                                                                    TBAS1820
  152   CONTINUE                                                                     TBAS1830
        RETURN                                                                       TBAS1840
        END                                                                          TBAS1850
```

```
      SUBROUTINE  WBASIC(JX,X,LXA,MXA,L,M,ID1,ID2,ID3,ICA,CA,INX         WBAS0010
     1,  TITLE,HEAD )                                                    WBAS0020
C  TO WRITE INDENTURED CODED TABLES                                      WBAS0030
C  PGM=NXX        (WITH CLASSIFICATION LABELS) VER.4 06-20-73 FORT.IV EBCD WBAS0040
C  L.C.GREGORY,UNIT 2-54240,EXT 510.                                     WBAS0050
C  SUBROUTINE PURPOSE, TO WRITE A SET OF BASIC LABELED TABLES.           WBAS0060
C  TITLE IS 15A4 = 60H          ID1 & ID2 ARE UNIQUE TABLE IDENT CODES.  WBAS0070
C  HEAD IS  A4,3A4,A4 = 20H,PLUS 8( 2A4)=64H, PLUS 8( 2A4)=64H.ALL=148H WBAS0080
C   INX = 0 , USE STANDARD HEADINGS                                      WBAS0090
C       = 1 , FURNISH COL HEADS AT CALL TIME                             WBAS0100
C  WRITES ARRAY JX=(ROW LABELS) SIDE BY SIDE WITH ARRAY  X=(DATA).       WBAS0110
C   ICA CONTAINS 4 INTEGER NAMES AND VALUES                              WBAS0120
C    CA CONTAINS 4 REAL     NAMES AND VALUES                             WBAS0130
     COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12               WBAS0140
      INTEGER  ZIP,ZCODE,TITLE                                           WBAS0150
      COMMON/INOUT/NLINE,NPAGE,PCODE(20),MISC(7),XMISC(7),ZIP,ZCODE(19)WBAS0160
     1      ,JRASH(20),TRASH(20),IR(8),IC(8),DUM(8),IDUM(8),NFLAG,NFLAG2WBAS0170
      DIMENSION   JX(LXA,5), X(LXA,MXA),ICA(16),CA(16)                   WBAS0180
      DIMENSION   TITLE(15), HEAD(37)                                    WBAS0190
C  FORMATS                                                               WBAS0200
 2000 FORMAT(1H1,71X, 4HPAGE,I4/6X,19A4,A3)                              WBAS0210
 2002 FORMAT(//6X,4HID1=, I8,8H,    ID2=,I8,8H,    ID3=,I4/               WBAS0220
     1           6X,10HTABLE NO. ,2I8,I4,2I5,10A4)                       WBAS0230
 2008 FORMAT(1H0,33HERROR IN SUBROUTINE WBASIC AT E1=,F8.2,              WBAS0240
     1          /6X,10HTABLE NO. ,2I8,I4,2I5,10A4/(6X,3HE2=10G12.5) )    WBAS0250
 2012 FORMAT(7X, 3HROW,7H   ID. , 12HNAME * * * *,5H DATA,6(4X,4HCOL.,   WBAS0260
     1  I3,3X)/( 40X,6(4X,4HCOL.,I3,3X)) )                               WBAS0270
 2014 FORMAT(7X, 3HROW,2X,A4,1X,3A4,1X,A4,6(4X,2A4,2X)/                  WBAS0280
     1  (40X,6(4X, 2A4, 2X)) )                                           WBAS0290
 2016 FORMAT(/5X,2I5,2H. ,3A4,I5,6G14.6)                                 WBAS0300
 2017 FORMAT(/5X,2I5,2H. ,3A4,I5,6G14.6/(40X,6G14.6))                    WBAS0310
 2018 FORMAT(/12X,7HINTEGER,3X,4(2X,2A4,A2,G13.6)/                       WBAS0320
     1        (12X,7HREAL    ,3X,4(2X,2A4,A2,G13.6)) )                   WBAS0330
 1    E1 = 1.0                                                           WBAS0340
      IF (L*M.LE.LXA*MXA.AND.L.GE.0.AND.M.GE.0)  GO TO 22                WBAS0350
 2    NFLAG = NFLAG + 1                                                  WBAS0360
      WRITE(N6,2008)E1,TITLE,ID1,ID2,ID3,ID4                             WBAS0370
      RETURN                                                             WBAS0380
 22   ASSIGN 36 TO KS                                                    WBAS0390
      IF (M.GT.6)  ASSIGN 38 TO KS                                       WBAS0400
      I = 0                                                              WBAS0410
      K3 = MIN0(2*M,24) + 5                                              WBAS0420
      K4 = MIN0(  M,12) - 1                                              WBAS0430
      INCR = (M-1)/6 + 2                                                 WBAS0440
      INCM = INCR*L + 8                                                  WBAS0450
      IF (NLINE+INCM.LE.63)                 GO TO 26                     WBAS0460
C**                                                                      WBAS0470
 24   CALL PAGE                                                          WBAS0480
C                                                                        WBAS0490
C                                                                        WBAS0500
 26   NLINE = NLINE + 6                                                  WBAS0510
      WRITE(N6,2002)ID1,ID2,ID3,TITLE                                    WBAS0520
       IF ( INX.GT.0)                       GO TO 30                     WBAS0530
      WRITE(N6,2012)(J,J=1,K4)                                           WBAS0540
      GO TO 32                                                           WBAS0550
 30     WRITE(N6,2014)(HEAD(K),K=1,K3)                                  WBAS0560
 32     IF (L.LE.0)                         GO TO 42                    WBAS0570
C**                                                                     WBAS0580
 34     I = I + 1                                                       WBAS0590
        GO TO KS,(36,38)                                                WBAS0600
 36     WRITE(N6,2016)I,(JX(I,J),J=1,5), (X(I,K),K=1,M)                 WBAS0610
        GO TO 40                                                        WBAS0620
 38     WRITE(N6,2017)I,(JX(I,J),J=1,5), (X(I,K),K=1,M)                 WBAS0630
 40     NLINE = NLINE + INCR                                            WBAS0640
        IF (I.GE.L)                         GO TO 42                    WBAS0650
        IF (NLINE + INCR.LE.61)             GO TO 34                    WBAS0660
        GO TO 24                                                        WBAS0670
C**                                                                     WBAS0680
 42     WRITE(N6,2018)ICA,CA                                            WBAS0690
        NLINE = NLINE + 3                                               WBAS0700
        RETURN                                                          WBAS0710
        END                                                             WBAS0720



        SUBROUTINE  UPDA(M3,M4,M5,M6,IXX)                               UPDA0010
C  PGM=NFM. L.D.G.           VER.1. 7-29-73.  IBM 370 FORTRAN IV    EBCD UPDA0020
C MODS. INDX11(180)=70.                                       2         UPDA0030
C  TO READ CHANGES TO BASIC TABLES IN NAMELIST FORMAT                   UPDA0040
        COMMON/DEVICE/N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12            UPDA0050
        COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                    UPDA0060
     1,IDUMB(72), NFLAG, NFLAG2                                         UPDA0070
        COMMON/INOU11/ JD11,ND11,NB11,INDX11(180,7)                     UPDA0080
CZAZA                                                                   UPDA0090
        INTEGER TITLA                                                   UPDA0100
        COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)   UPDA0110
      1, TITLA(15),HEADA(37),IFMT                                       UPDA0120
      2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                         UPDA0130
        DIMENSION  BZA( 555)                                            UPDA0140
        EQUIVALENCE  (NCBA,BZA(1))                                      UPDA0150
        DATA   NC3,LZ,MZ,LA,MA/ 1,555,12,27,12/                         UPDA0160
        INTEGER  TITLE,ROWNAM                                           UPDA0170
        DIMENSION  TITLE(10),ROWNAM(3),DATA(12)                         UPDA0180
      1,      ICONST(4),CONST(4)                                        UPDA0190
        DATA DFAULT,IFAULT/1.E-66, 12345678/                            UPDA0200
        EQUIVALENCE (TITLA(6),TITLE(1)),(ICONST(1),IC1),(ICONST(2),IC2), UPDA0210
      1   (ICONST(3),IC3),(ICONST(4),IC4),(CONST(1),RC1),(CONST(2),RC2), UPDA0220
      2   (CONST(3),RC3),(CONST(4),RC4)                                 UPDA0230
        NAMELIST/NAMUP/ID1,ID2,ID3,TITLE,ROWSEQ,ROWNAM,CODE,DATA        UPDA0240
      1,          IC1,IC2,IC3,IC4,     RC1,RC2,RC3,RC4,RETURN           UPDA0250
      2,      ICONST,CONST                                              UPDA0260
        DATA   BLANK,LANK/'    ','    '/,JXXX/'XXXX'/                   UPDA0270
C***   SAMPLE INPUT                                                     UPDA0280
CIP 11 1 2  6210        -1                  UPDATE ORIGINAL TABLE.      UPDA0290
C&NAMUP ROWSEQ=10., DATA(2) = 222.,333.,DATA(7)=7.,               &END  UPDA0300
C&NAMUP ROWSEQ=14., CODE=6253., DATA(3)= 3.,IC2= 2,    RETURN=1., &END  UPDA0310
CIP  8                                      EXECUTE                     UPDA0320
CIP 11 2 2  6210        -1                  UPDATE TABLE UPDATE         UPDA0330
C       ETC.                                                            UPDA0340
C***                                                                    UPDA0350
```

```
 2000   FORMAT(1H0,'NO TABLE FOUND TO UPDATE FOR ID= ',2I10)                  UPDA0360
 2002   FORMAT(1H0,'TABLE ALREADY FULL FOR ID=',2I10,',  ROWSEQ=',F4.0)       UPDA0370
 2004   FORMAT(1H0,'EXTRA LINE ADDED TO TABLE FOR ID=',2I10,                  UPDA0380
     1        ',  ROWSEQ=',F4.0)                                              UPDA0390
 2006   FORMAT(5X, I5,2H. ,3A4,I5, 6G14.6/40X,6G14.6)                         UPDA0400
 2008   FORMAT(1H0,'NEW LINE FOR TABLE WITH ID=',2I10)                        UPDA0410
 2043   FORMAT(1H0,'CONSTANTS UPDATED',3X,4(2X,2A4,A2,G13.6)                  UPDA0420
     1       / 21X, 4(2X,2A4,A2,G13.6))                                       UPDA0430
C***                                                                          UPDA0440
 1      KZARL = 96 + LA*(5 + MA)                                              UPDA0450
        IXX = 0                                                               UPDA0460
        RETURN = 0.                                                           UPDA0470
        IPRINT = M4                                                           UPDA0480
        ID1    = M5                                                           UPDA0490
        ID2    = M6                                                           UPDA0500
        ID3    = -1                                                           UPDA0510
 2      NB1 = NB11 + 1                                                        UPDA0520
        DO 10 J=1,NB11                                                        UPDA0530
        I   = NB1 - J                                                         UPDA0540
        IF (INDX11(I,1).NE.1)  GO TO 10                                       UPDA0550
        IF (INDX11(I,2).EQ.ID1.AND.INDX11(I,3).EQ.ID2)  GO TO 4               UPDA0560
        GO TO 10                                                              UPDA0570
 4      IF (M3.GT.1)  GO TO 14                                                UPDA0580
        IF (INDX11(I,7).GE.0)  GO TO 14                                       UPDA0590
 10     CONTINUE                                                              UPDA0600
C***  NO TABLE FOUND                                                          UPDA0610
 12     IXX = IXX + 1                                                         UPDA0620
        WRITE(N6,2000) ID1,ID2                                                UPDA0630
        NLINE = NLINE + 2                                                     UPDA0640
 13     IF (RETURN.GT..01)  RETURN                                            UPDA0650
        READ (N5,NAMUP)                                                       UPDA0660
        GO TO 13                                                              UPDA0670
C***  TABLE FOUND                                                             UPDA0680
 14     JB  = I                                                               UPDA0690
        JD11 = INDX11(JB,4)                                                   UPDA0700
        READ (N11'JD11) BZA                                                   UPDA0710
C***  READY FOR UPDATE                                                        UPDA0720
 20     DO 22 I=1,12                                                          UPDA0730
 22     DATA(I)=DFAULT                                                        UPDA0740
        ROWSEQ=0.                                                             UPDA0750
        CODE=DFAULT                                                           UPDA0760
        DO 23 I=1,4                                                           UPDA0770
        ICONST(I)=IFAULT                                                      UPDA0780
 23     CONST(I)=DFAULT                                                       UPDA0790
        ROWNAM(1) = JXXX                                                      UPDA0800
        ROWNAM(2) = LANK                                                      UPDA0810
        ROWNAM(3) = LANK                                                      UPDA0820
C***  READ NAMELIST                                                           UPDA0830
        READ (N5,NAMUP)                                                       UPDA0840
        KSEQ = ROWSEQ + .01                                                   UPDA0850
        KODE = CODE + .01                                                     UPDA0860
        DO 28 I=1,LA1                                                         UPDA0870
        IF (KA1(I,1).EQ.KSEQ)  GO TO 33                                       UPDA0880
 28     CONTINUE                                                              UPDA0890
        IF (LA1.LT.LA)  GO TO 29                                              UPDA0900
```

```
      IXX = IXX + 1                                                     UPDA0910
      WRITE(N6,2002) ID1, ID2, ROWSEQ                                   UPDA0920
      GO TO 13                                                          UPDA0930
C***  ADD LINE TO UPDATE                                                UPDA0940
 29   LA1 = LA1 + 1                                                     UPDA0950
      KA1(LA1,1) = KSEQ                                                 UPDA0960
      DO 30 I=1,3                                                       UPDA0970
 30   KA1(LA1,I+1) = ROWNAM(I)                                          UPDA0980
      KA1(LA1,5)   = KODE                                               UPDA0990
      DO 31 I=1,MA1                                                     UPDA1000
 31   VA1(LA1,I) = DATA(I)                                              UPDA1010
C***  PRINT                                                             UPDA1020
      IF (NLINE.GT.55)  CALL PAGE                                       UPDA1030
      WRITE(N6,2004) ID1, ID2, ROWSEQ                                   UPDA1040
      JROW = LA1                                                        UPDA1050
      GO TO 42                                                          UPDA1060
C***  FOUND ROW.  TEST FOR CODE                                         UPDA1070
 33   JROW = I                                                          UPDA1080
      IF(CODE.NE.DFAULT) KA1(JROW,5)=KODE                               UPDA1090
C***  TEST FOR DATA                                                     UPDA1100
 36   DO 40 I=1,MA1                                                     UPDA1110
      AD=DATA(I)                                                        UPDA1120
      IF(AD.NE.DFAULT) VA1(JROW,I)=AD                                   UPDA1130
 40   CONTINUE                                                          UPDA1140
      IF (NLINE.GT.55)  CALL PAGE                                       UPDA1150
      WRITE(N6,2008) ID1, ID2                                           UPDA1160
 42   WRITE(N6,2006) (KA1(JROW,J),J=1,5),(VA1(JROW,J),J=1,MA1)          UPDA1170
      NLINE = NLINE + 4                                                 UPDA1180
      J=0                                                               UPDA1190
      DO 430 I=1,4                                                      UPDA1200
      K=ICONST(I)                                                       UPDA1210
      IF(K.EQ.IFAULT) GO TO 43                                          UPDA1220
      ICA(I*4)=K                                                        UPDA1230
      J=1                                                               UPDA1240
 43   AD=CONST(I)                                                       UPDA1250
      IF(AD.EQ.DFAULT) GO TO 430                                        UPDA1260
      CA(I*4)=AD                                                        UPDA1270
      J=1                                                               UPDA1280
 430  CONTINUE.                                                         UPDA1290
      IF(J.EQ.0) GO TO 44                                               UPDA1300
      WRITE(N6,2043) ICA,CA                                             UPDA1310
      NLINE=NLINE+3                                                     UPDA1320
C***  READ MORE CHANGES                                                 UPDA1330
 44   IF (RETURN.LE..01)  GO TO 20                                      UPDA1340
C***  WRITE UPDATE ON DISK 11                                           UPDA1350
      NB1 = NB11 + 1                                                    UPDA1360
      DO 46 J=1,NB11                                                    UPDA1370
      I   = NB1 - J                                                     UPDA1380
      IF (INDX11(I,1).NE.1)   GO TO 46                                  UPDA1390
      IF (INDX11(I,2).EQ.ID1.AND.INDX11(I,3).EQ.ID2)  GO TO 48          UPDA1400
 46   CONTINUE                                                          UPDA1410
      GO TO 12                                                          UPDA1420
 48   JB  = I                                                           UPDA1430
      IF ( INDX11(JB,7).GE.0)  GO TO 50                                 UPDA1440
      IDA3 = ID3                                                        UPDA1450
      JD11 = INDX11(JB,4)
      WRITE(N11'JD11) BZA
      GO TO 52
C***  CREATE TEMPORARY FILE (ID3=-1)
 50   KZARL = LZ
      IDA3 = ID3
      CALL WCBD11(BZA,LZA,MZA,NCBA,KZARL,ID1,ID2,ID3, IX, IY )
C***  PRINT OPTION
 52   IF (IPRINT.LE.1)  GO TO 60
      IF (IPRINT.LE.2)  GO TO 54
      IY   = 3
      CALL RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IX)
      IF ( IX.EQ.0)  GO TO 54
      GO TO 12
 54   IP   = 1
      CALL WBASIC(KA1,VA1,LAA1,MAA1,LA1,MA1,IDA1,IDA2,ICA3,ICA,CA,IP,
     1           TITLA,HEADA)
 60   RETURN
      END


      SUBROUTINE  STRING(M3,M4,M5,M6,N5,N6,N8,NCODE)
C  TO RECORD, RETRIEVE, OR PRINT STRINGS ON TAPE 8)      10-28-74
C  FOR NFM AT DALLAS. NEEDS 1.MODS TO MAIN  2. TAPE 8.
CALLS
CXX 12 1 0      74000     1        READ TO XXX,PUT ON TAPE 8,CASE=7400,NW=1.
CXX                                END OF STRING)
CXX 12 2 0      74000              FIND CASE 74000 ON TAPE 8,INSERT HERE.
CXX 12 3 0      74000              PRINT CASE 74000,ALL CASES IF CASE=0
CFORMATS
 1000  FORMAT(20A4)
 1001  FORMAT(  A4)
 2000  FORMAT(1X,I4,1H.,20A4)
 1002  FORMAT( A4, 6X, I10,20X,10A4)
 2001  FORMAT( 4HCASE,6X,I10,20X,10A4)
 2003  FORMAT(54HXXX 12 1-1                                   END OF STRING,
     1     26X)
 2004  FORMAT(4HEND ,76X)
 2005  FORMAT( 6X, 5HCASE=,I10,16H,     NOW ON TAPE8)
 2006  FORMAT( 6X, 5HCASE=,I10, 9H,     FOUND ,6X,10A4)
 2007  FORMAT( 6X, 5HCASE=,I10,13H,     NOT FOUND )
CNOTES.
C  1.THE FIRST CARD AFTER A XXX 12 1 0  MUST BE A ZIP CARD.
C  2.THE NEXT  CARD AFTER A XXX  (END OF STRING)  MUST BE A ZIP CARD.
C.  3.Z6=1 USED IN XXX 12 1 0 MEANS WRITE STARTING AT FRONT END OF TAPE8
C  4.CAN LOAD A CASE AND USE IN SAME JOB IF LATER USE A XXX 12 2 CARD.
C  5.PRINT OF CASE DETAILS APPEAR ONLY FIRST TIME PUT ON TAPE8, AND
C     THEN ONLY IN INPUT CARD IMAGES(IF TURNED ON).
C  6.CASE DETAILS PRINTED BY XXX 12 3, ALL CASES IF CASE=0. IF DESIRED,
C     BEST TO PRINT BEFORE USING A  XXX 12 2 CARD, SINCE THIS MEANS
C     FIND AND USE IMMEDIATELY.
C  7.IF CANNOT FIND A CASE FOR XXX 12 2 , PROGRAM STOPS,XXX 12 3 GOES ON
C  8.IF STRING CONTAINS TABLE UPDATES, TABLES MUST BE AHEAD OF UPDATES,
C     EITHER IN THIS STRING OR IN PREVIOUS STRING, OR REGULAR STREAM.
```

```
C   9.ON XXX 12 1 CARD USE COLS 42-80 FOR STRING TITLE.                  STNG0340
       DIMENSION CARD(20),NCODE(19),TITLE(10)                            STNG0350
       DATA  CASE, XXX, END/4HCASE,4HXXX ,4HEND /                        STNG0360
 1     IF (M3.GT.1)  GO TO 20                                            STNG0370
       NUTAPE = M6                                                       STNG0380
       IF (NUTAPE.EQ.1)  GO TO 10                                        STNG0390
C   SEARCH FOR END                                                       STNG0400
 6     READ (N8,1001) TEST                                               STNG0410
       IF (TEST.NE.END)  GO TO 6                                         STNG0420
       BACKSPACE  N8                                                     STNG0430
 10    WRITE(N8,2001) M5,(NCODE(I),I=10,19)                              STNG0440
 12    READ (N5,1000) CARD                                               STNG0450
       IF (CARD(1).EQ.XXX)  GO TO 14                                     STNG0460
       WRITE(N8,1000) CARD                                               STNG0470
       GO TO 12                                                          STNG0480
 14    WRITE(N8,2003)                                                    STNG0490
       WRITE(N8,2004)                                                    STNG0500
       END FILE  N8                                                      STNG0510
       REWIND    N8                                                      STNG0520
       WRITE(N6,2005) M5                                                 STNG0530
       RETURN                                                            STNG0540
C   FIND CASE= M5                                                        STNG0550
 20    IF (M3.GT.2)  GO TO 30                                            STNG0560
 21    READ (N8,1002) TEST, NUM, TITLE                                   STNG0570
       IF (TEST.EQ.END)  GO TO 26                                        STNG0580
       IF (NUM.EQ. M5 )  GO TO 24                                        STNG0590
 22    READ (N8,1001) TEST                                               STNG0600
       IF (TEST.EQ.XXX)  GO TO 21                                        STNG0610
       GO TO 22                                                          STNG0620
 24    IF (M3.EQ.3)  GO TO 32                                            STNG0630
       WRITE(N6,2006) M5, TITLE                                          STNG0640
       N5  = N8                                                          STNG0650
       RETURN                                                            STNG0660
 26    WRITE(N6,2007) M5                                                 STNG0670
       IF (M3.EQ.3)  RETURN                                              STNG0680
       STOP                                                              STNG0690
C   PRINT CASE(S) ON TAPE 8, ALL IF CASE NUMBER IS ZERO                  STNG0700
 30    REWIND  N8                                                        STNG0710
       IF (M5.LE. 0)  GO TO 36                                           STNG0720
       GO TO 21                                                          STNG0730
 32    CALL  PAGE                                                        STNG0740
       WRITE(N6,2006) NUM, TITLE                                         STNG0750
       K  = 0                                                            STNG0760
 34    READ (N8,1000) CARD                                               STNG0770
       K  = K + 1                                                        STNG0780
       WRITE(N6,2000) K,CARD                                             STNG0790
       IF (CARD(1).NE.XXX)  GO TO 34                                     STNG0800
       IF (M5.GT.0)  GO TO 38                                            STNG0810
 36    READ (N8,1002) TEST, NUM, TITLE                                   STNG0820
       IF (TEST.NE.END)  GO TO 32                                        STNG0830
 38    REWIND N8                                                         STNG0840
       RETURN                                                            STNG0850
       END                                                               STNG0860
```

```
      SUBROUTINE  CARD(N5,N6,NX, CLAS,PCODE)
C  PGM=NXX(NEM).  L.D.G.          VER.1  7-23-73        FORTRAN IV      EBCD
C  TO READ CARD IMAGES FROM N5, COPY ONTO NX,  AND PRINT ON N6
 1000  FORMAT(20A4)
 1001  FORMAT(1X,19A4,A3)
 2000  FORMAT(1X,I5,'.',20A4)
 2001  FORMAT(1X,I5,'.',1X,19A4,A3)
 2002  FORMAT(6X,'INPUT CARD IMAGES',/)
       DIMENSION  KARD(20),CLAS(20),PCODE(20)
       DATA  KZIP,K10/'ZIP ','1C  '/
       K1    = 1
       KCARD = 2
       NLINE = 2
       WRITE(N6,2002)
       WRITE(NX, 1001) CLAS, PCODE
       WRITE(N6,2001) K1, CLAS, KCARD, PCODE
       GO TO 2
 1     NLINE = 0
       WRITE(N6,2002)
 2     NLINE = NLINE + 1
       KCARD = KCARD + 1
       READ (N5,1000) KARD
       WRITE(NX, 1000)KARD
       WRITE(N6,2000) KCARD, KARD
       IF (KARD(1).EQ.KZIP.AND.KARD(2).EQ.K10)  GO TO 8
       IF (NLINE.LT.50)  GO TO 2
       CALL PAGE
       GO TO 1
 8     END FILE NX
       REWIND   NX
       RETURN
       END


      SUBROUTINE UNIQUE(KA,KU,IU)
C**PGM=NXX. L.D.G.               9-7-73   FORTRAN IV                  EBCDIC
C  FOR ITEMS IN KA(KU)  IN INCREASING OR DECREASING ORDER, OR WITH
C  ALL LIKE ITEMS ADJACENT, PUT A UNIQUE LIST IN THE FRONT OF KA AND
C  RETURN THE REDUCED NUMBER  IU,(IU.LE.KU).
C**
      DIMENSION KA(KU),RA(KU)
      IF(1-KU) 8,6,4
 4    IU=0
      RETURN
 6    IU=1
      RETURN
 8    IU=1
      K1=KU-1
      I=1
 10   K=KA(I)
      I1=I+1
      CO 12 J=I1,KU
      IF(K.EQ.KA(J)) GO TO 12
      IU=IU+1
      KA(IU)=KA(J)                                                      UNIQ0210
      I=J-1                                                             UNIQ0220
      GO TO 16                                                          UNIQ0230
   12 CONTINUE                                                          UNIQ0240
       RETURN                                                           UNIQ0250
   16 I=I+1                                                             UNIQ0260
      IF(I.LE.K1) GO TO 10                                              UNIQ0270
      RETURN                                                            UNIQ0280
C***                                                                    UNIQ0290
C                                                                       UNIQ0300
       ENTRY       RNIQUE(RA,KU,IU,E)                                   UNIQ0310
C***                                                                    UNIQ0320
       IF(1-KU) 28,26,24                                                UNIQ0330
   24 IU=0                                                              UNIQ0340
      RETURN                                                            UNIQ0350
   26 IU=1                                                              UNIQ0360
      RETURN                                                            UNIQ0370
   28 IU=1                                                              UNIQ0380
      K1=KU-1                                                           UNIQ0390
      I=1                                                               UNIQ0400
   30  R=RA(I)                                                          UNIQ0410
      I1=I+1                                                            UNIQ0420
      DO 32 J=I1,KU                                                     UNIQ0430
       IF (R.LE.(RA(J)+E).AND.R.GE.(RA(J)-E))    GO TO 32               UNIQ0440
      IU=IU+1                                                           UNIQ0450
      RA(IU)=RA(J)                                                      UNIQ0460
      I=J-1                                                             UNIQ0470
      GO TO 36                                                          UNIQ0480
   32  RA(J) = R                                                        UNIQ0490
       RETURN                                                           UNIQ0500
   36 I=I+1                                                             UNIQ0510
      IF(I.LE.K1) GO TO 30                                              UNIQ0520
      RETURN                                                            UNIQ0530
      END                                                               UNIQ0540


       SUBROUTINE  SETUPA(IPRINT)                                       SETA0010
C  PGM=NXX.   L.D.GREGORY. VER.8   9-15-73   FORTRAN IV. EBCD.          SETA0020
C  TO SET UP INITIAL POSITIONS AND PLANNED ROUTES                       SETA0030
CALLEC BY  MAIN AND ZIP CARD AS FOLLOWS                                 SETA0040
CIP  8 1 P                             SET UP NAV.INIT.PKG/PRINT P=1,2SETA0050
C  NOTES, IN CSETA, RED & BLU FOLLOW PATTERN AS BELOW,           /BLU   SETA0060
C    1.NRG = NO.OF RED GROUPS                                    /NBG   SETA0070
C    2.KRGN(I) = ITH GROUP NO.,E.G.        2180000  (FROM TABLE 0)/KBGN SETA0080
C    3.KRGK(I) = ITH GROUP KODE,E.G.          1118  (FROM TABLE 2)/KBGK SETA0090
C               WHICH RELATES RED GROUP 11 TO 18                 /      SETA0100
C    4.RGC( 8,6,K) = KTH. TABLE OF RED GROUP CENTERS VS TIME     /BGC   SETA0110
C    5.NRU(I) = NO.OF RED UNITS IN ITH GROUP + 100* STARTING LINE/NBU   SETA0120
C    6.KRUK( ,J) = RED UNIT KODE,E.G.      2010100, 83410000, ETC /KBUK SETA0130
C    7.RA,RB,RC,RD = RED CENTER,VECTOR,START,RENDEZVOUS POINTS.  /BA,BB, SETA0140
C**                                                                     SETA0150
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             SETA0160
       COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                     SETA0170
     1,ICUMB(72), NFLAG, NFLAG2                                         SETA0180
```

```
CZAZA
      INTEGER TITLA
      COMMON/CZAZA/NCBA,IDA1, IDA2, IDA3,JZARL,LZA,MZA,ICA(16),CA(16)
     1, TITLA(15),HEADA(37),IFMT
     2, LA1,LAA1,MA1,MAA1,KA1( 27,5),VA1(27,12)
      DIMENSION  BZA( 555)
      FQUIVALENCF  (NCBA,BZA(1))
      DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/
      COMMON/CWORK/KA2(27),VA2(27,12),KA3(27),VA3(27,12),
     1            LA2,MA2,               LA3,MA3              ,PWORK(794)
      DIMENSION  KWORK(706)
      FQUIVALENCF  (KA2(1),KWCRK(1))
CNAVIG
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,
     1    NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),
     2    NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),
     3          TTIME,NUMX,
     4    KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),
     5    KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)
      DATA  RAD/0.0174533/,PI/3.14159/
      DIMENSION  VA(12)
      NAMELIST/NAMA1/NBG,KBGN,KBGK
      NAMELIST/NAMA2/NRG,KRGN,KRGK
      NAMELIST/NAMA3/KBU,NBU
      NAMELIST/NAMA4/KRU,NRU
      NAMELIST/NAMA5/KBU,BREL
      NAMELIST/NAMA6/KRU,RREL
 2005 FORMAT(1HO,22HERROR IN SETUPA AT E1=,F8.2,3X,'ID=',2I10)
 2006 FORMAT(1HO,  'CHECK FOR MISSING TABLE, ID=',2I10)
 2C08 FORMAT(/6X,' AFTER INITIAL SETUP',/6X,' UNIT  NAME',10X,'CODF',
     1 6X,'TYPE GRUP STAT   DELTA X   DELTA Y   DELTA Z    VALUE')
 2009 FORMAT(/6X,I4,'.  ',2A4,2I10,2I5, 4F10.3)
C** ARITHMETIC FUNCTIONS,
      JUPAK(N,KD,KM) = MOD( N/10**KD, 10**KM)
      HRS(X)   = FLOAT(IFIX(X)) + (X-FLOAT(IFIX(X))) / 0.6
C**
 1    KZARL = 96 + LA*(5 + MA)
      JZARL = KZARL
      NFLAG2= 0
      NGMAX = NGMX
      NUMAX = NUMX
      ID2   = 0
C** FIND TABLE OF TABLES. TABLE NO.= 0 .
      KTRAN = 1
      ID1   = 00010000
 3    IY  = 1
 4    CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX.EQ.0)  GO TO (10,22,52), KTRAN
      E1   = 4.03
 6    WRITE(N6,2006)      ID1, ID2
      NFLAG2 = NFLAG2 + 1
      GO TO ( 8,50,70), KTRAN
 8    STOP
C** STORE TABLE NUMBERS FROM TABLE NO.= 0 .
 10   NBG = 0
```

```
      NRG = 0                                                           SETA0740
      DO 14  I=1,LA1                                                    SETA0750
      DO 14  J=1,MA1                                                    SETA0760
      K    = (VA1(I,J) + .001)                                          SETA0770
      IF (K.LE.0.OR. K.EQ.100.OR. K.EQ.200.OR. K.GE.300)   GO TO 14     SETA0780
      IF (K.GT.200)   GO TO 12                                          SETA0790
      NBG = NBG + 1                                                     SETA0800
      KBGN(NBG) = K * 10000                                             SETA0810
      GO TO 14                                                          SETA0820
 12   NRG = NRG + 1                                                     SETA0830
      KRGN(NRG) = K * 10000                                             SETA0840
 14   CONTINUE                                                          SETA0850
C** SORT TABLE NUMBERS                                                   SETA0860
      CALL  KORDER(KBGN,NBG)                                            SETA0870
      CALL KORDER(KRGN,NRG)                                             SETA0880
C** START BLU SET UP. GET BLU GROUP CENTERS FROM TABLE NO.=1000000       SETA0890
 20   KTRAN  = 2                                                        SETA0900
      ID1  = 1000000                                                    SETA0910
      GO TO 3                                                           SETA0920
 22   BA   = ICA(4)                                                     SETA0930
      BB   = ICA(8)                                                     SETA0940
      BC   = ICA(12)                                                    SETA0950
      BD   = ICA(16)                                                    SETA0960
      BE   = CA(12)                                                     SETA0970
      BF   = CA(16)                                                     SETA0980
 24   DO 28  I=1,LA1                                                    SETA0990
      KA2(I) = MOD(KA1(I,5),100) * 10000 + ID1                          SETA1000
      DO 26  J=1,MA1                                                    SETA1010
 26   VA2(I,J) = VA1(I,J)                                               SETA1020
      VA2(I,2) = VA2(I,2) / 6.080                                       SETA1030
      VA2(I,3) = HRS(VA2(I,3))                                          SETA1040
      DO 28  J= 4,10,3                                                  SETA1050
      VA2(I,J+2) = VA2(I,J+2) / 6.080                                   SETA1060
 28   VA2(I,J)    = HRS(VA2(I,J) )                                      SETA1070
      LA2 = LA1                                                         SETA1080
      MA2 = MA1                                                         SETA1090
C** CYCLE THRU BLU GROUP CENTERS IN TABLE NO.=1000000                    SETA1100
 32   IXX = 0                                                           SETA1110
      KK   = 0                                                          SETA1120
      DO 38  KG=1,NBG                                                   SETA1130
      KGN = KBGN(KG)                                                    SETA1140
      DO 36  KH=1,LA2                                                   SETA1150
      IF (KGN.NE.KA2(KH))   GO TO 36                                    SETA1160
      KK   = KK + 1                                                     SETA1170
      KBGN(KK) = KGN                                                    SETA1180
      KBGK(KK) = KA1(KH,5)                                              SETA1190
      DO 34  KJ=1,12                                                    SETA1200
 34   VA(KJ) = VA2(KH,KJ)                                               SETA1210
      CALL   NAVIT(BGC,NGMAX,KK,VA,ICA,CA,IXY,N6)                       SETA1220
      IXX = IXX + IXY                                                   SETA1230
 36   CONTINUE                                                          SETA1240
 38   CONTINUE                                                          SETA1250
      NBG = KK                                                          SETA1260
      IF (IXX.NE.0)   NFLAG2 = NFLAG2 + 1                               SETA1270
C** START RED SET UP. GET RED GROUP CENTERS FROM TABLE NO.=2000000       SETA1280
```

```
 50   KTRAN = 3
      ID1 = 2000000
      GO TO 3
 52   RA  = ICA(4)
      RB  = ICA(8)
      RC  = ICA(12)
      RD  = ICA(16)
      RE  = CA(12)
      RF  = CA(16)
 54   DO 58 I=1,LA1
      KA3(I) = MOD(KA1(I,5),100) * 10000 + ID1
      DO 56 J=1,MA1
 56   VA3(I,J) = VA1(I,J)
      VA3(I,2) = VA3(I,2) / 6.080
      VA3(I,3) = HRS(VA3(I,3))
      DO 58 J= 4,10,3
      VA3(I,J+2) = VA3(I,J+2) / 6.080
 58   VA3(I,J)    = HRS(VA3(I,J) )
      LA3 = LA1
      MA3 = MA1
C** CYCLE THRU RED GROUP CENTERS IN TABLE NO.=2000000
 62   IXX = 0
      KK  = 0
      DO 68 KG=1,NRG
      KGN = KRGN(KG)
      DO 66 KH=1,LA3
      IF (KGN.NE.KA3(KH))  GO TO 66
      KK  = KK + 1
      KRGN(KK) = KGN
      KRGK(KK) = KA1(KH,5)
      DO 64 KJ=1,12
 64   VA(KJ) = VA3(KH,KJ)
      CALL  NAVIT(RGC,NGMAX,KK,VA,ICA,CA,IXY,N6)
      IXX = IXX + IXY
 66   CONTINUE
 68   CONTINUE
      NRG = KK
      IF (IXX.NE.0)  NFLAG2 = NFLAG2 + 1
C** PRINT OPTION
 70   IF (IPRINT.LE.0)  GO TO 78
      IF (NBG.LE.0)  GO TO 74
      WRITE(N6,NAMA1)
      NLINE = NLINE + 6
      IF (IPRINT.LE.1)  GO TO 74
      DO 72 I=1,NBG
      CALL  WMAT3(2,ICA,BGC, 8, 6,NGMAX, 8, 6, I,NLINE,
    1 60HBORDERED TABLE. BLU GROUP CENTERS VS TIME                       )
 72   CONTINUE
 74   IF (NRG.LE.0)  GO TO 78
      CALL  PAGE
      WRITE(N6,NAMA2)
      NLINE = NLINE + 6
      IF (IPRINT.LE.1)  GO TO 78
      DO 76 I=1,NRG
      CALL  WMAT3(2,ICA,RGC, 8, 6,NGMAX, 8, 6, I,NLINE,
     1 60HBORDERED TABLE. RED GROUP CENTERS VS TIME                           ) SETA1840
 76     CONTINUE                                                                SETA1850
CX* SET UP UNITS IN  KBUK AND KRUK  (BLU & RED)                                 SETA1860
C   CODE  NBU(I) =  100*J + K, WHERE K=NO.UNITS IN GROUP I                      SETA1870
C                                     J=STARTING LINE IN KBUK                   SETA1880
C   KBUK(1,J)  = UNIT KODE,E.G.  1010100  FOR  JTH  UNIT                        SETA1890
C   KBUK(2,J)  = UNIT TYPE,E.G. 61130000                                        SETA1900
C   KBUK(3,J)  = BACKWARD REF TO GROUP LINE NO. IN KBGN,BGC,KRGN,RGC            SETA1910
C   KBUK(4,J)  = STATUS         ALIVE = 1                                       SETA1920
C**    ADD THE FOLLOWING TO GROUP POS TO GET UNIT POS                           SETA1930
C   BREL(1,J)  = DELTA  X  FOR BLU.      RREL(1,J)  = DELTA  X  FOR RED         SETA1940
C   BREL(2,J)  =        Y          .     RREL(2,J)  =        Y                  SETA1950
C   BREL(3,J)  =        Z          .     RREL(3,J)  =        Z                  SETA1960
C   BREL(4,J)  = VALUE             ,     RREL(4,J)  = VALUE                     SETA1970
C   BV(J, )    = ALT & VEL         ,     RV(J, )    = ALT & VEL                 SETA1980
C   KBU        = TOTAL NO. OF BLU UNITS                                         SETA1990
C** CYCLE THRU GROUP TABLES & SET UP UNITS                                       SETA2000
 78     IF (NBG.LE.0)  GO TO 100                                                SETA2010
C** BLU UNITS                                                                    SETA2020
 80     KBU = 0                                                                 SETA2030
        ID2 = 0                                                                 SETA2040
        IY  = 1                                                                 SETA2050
        DO  99 KG=1,NBG                                                         SETA2060
        ID1 = KBGN(KG)                                                          SETA2070
        CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                        SETA2080
        IF ( IXX.EQ.0)  GO TO 84                                                SETA2090
 81     E1  = 81.00                                                             SETA2100
 82     WRITE(N6,2006)      ID1, ID2                                            SETA2110
        NFLAG2 = NFLAG2 + 1                                                     SETA2120
        GO TO 99                                                                SETA2130
C** CYCLE THRU UNITS IN BLU GROUP TABLE                                          SETA2140
 84     CALL  UNISET(KA1,VA1,LA,MA,LA1,MA1,NGMAX,KG,NUMAX,NBU,KBU,NAMBU,        SETA2150
     1                KBUK,BREL,BV,ID1,IXX)                                     SETA2160
        IF ( IXX.EQ.0)  GO TO 99                                                SETA2170
        NFLAG2 = NFLAG2 + 1                                                     SETA2180
 99     CONTINUE                                                                SETA2190
C**  RED UNITS                                                                   SETA2200
 100    IF (NRG.LE.0)  GO TO 120                                                SETA2210
        KRU = 0                                                                 SETA2220
        ID2 = 0                                                                 SETA2230
        IY  = 1                                                                 SETA2240
        DO 116 KG=1,NRG                                                         SETA2250
        ID1 = KRGN(KG)                                                          SETA2260
        CALL RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                         SETA2270
        IF ( IXX.EQ.0)  GO TO 114                                               SETA2280
 111    E1  = 111.00                                                            SETA2290
 112    WRITE(N6,2006)      ID1, ID2                                            SETA2300
        NFLAG2 = NFLAG2 + 1                                                     SETA2310
        GO TO 116                                                               SETA2320
C**  CYCLE THRU UNITS IN RED GROUP TABLE                                         SETA2330
 114    CALL UNISET(KA1,VA1,LA,MA,LA1,MA1,NGMAX,KG,NUMAX,NRU,KRU,NAMRU,         SETA2340
     1                KRUK,RREL,RV,ID1,IXX)                                     SETA2350
        IF ( IXX.EQ.0)  GO TO 116                                               SETA2360
        NFLAG2 = NFLAG2 + 1                                                     SETA2370
 116    CONTINUE                                                                SETA2380
```

```
C**  CHANGE VALUES  IN  BREL  TO DELTA   X, Y, Z, REF HEADING AT T=0      SETA2390
 120    IF (KBU.LE.0)   GO TO 124                                          SETA2400
        DO 122 KU=1,KBU                                                    SETA2410
        KG   = KBUK(3,KU)                                                  SETA2420
        HEAD = AINT(BGC(6,5,KG) )                                          SETA2430
        BEAR = (HEAD + BREL(3,KU)) * RAD                                   SETA2440
        RG    = BREL(2,KU)                                                 SETA2450
        BREL(1,KU) = RG * SIN(BEAR)                                        SETA2460
        BREL(2,KU) = RG * COS(BEAR)                                        SETA2470
 122    BREL(3,KU) = .0                                                    SETA2480
C**  PRINT OPTION                                                          SETA2490
        IF (IPRINT.LT.2)  GO TO 124                                        SETA2500
        CALL  PAGE                                                         SETA2510
        WRITE(N6,NAMA3)                                                    SETA2520
        WRITE(N6,2008)                                                     SETA2530
        DO 1221 J = 1, KBU                                                 SETA2540
 1221   WRITE(N6,2009) J,(NAMBU(J,I),I=1,2),(KBUK(I,J),I=1,4),             SETA2550
      1    (BREL(I,J),I=1,4)                                               SETA2560
C**  CHANGE VALUES  IN   RREL  TO DELTA   X, Y, Z, REF HEADING AT T=0     SETA2570
 124    IF (KRU.LE.0)   GO TO 128                                          SETA2580
        DO 126 KU=1,KRU                                                    SETA2590
        KG   = KRUK(3,KU)                                                  SETA2600
        HEAD = AINT(RGC(6,5,KG) )                                          SETA2610
        BEAR = (HEAD + RREL(3,KU)) * RAD                                   SETA2620
        RG    = RREL(2,KU)                                                 SETA2630
        RREL(1,KU) = RG * SIN(BEAR)                                        SETA2640
        RREL(2,KU) = RG * COS(BEAR)                                        SETA2650
 126    RREL(3,KU) = .0                                                    SETA2660
C**  PRINT OPTION                                                          SETA2670
        IF (IPRINT.LT.2)  GO TO 128                                        SETA2680
        CALL  PAGE                                                         SETA2690
        WRITE(N6,NAMA4)                                                    SETA2700
        WRITE(N6,2008)                                                     SETA2710
        DO 1261 J = 1, KRU                                                 SETA2720
 1261   WRITE(N6,2009) J,(NAMRU(J,I),I=1,2),(KRUK(I,J),I=1,4),             SETA2730
      1    (RREL(I,J),I=1,4)                                               SETA2740
 128    IF (NFLAG2.NE.0)       NFLAG=NFLAG + 1                             SETA2750
        IF (NBG*NRG*KBU*KRU  .LE. 0) NFLAG = NFLAG + 1                     SETA2760
        RETURN                                                             SETA2770
        END                                                                SETA2780


      SUBROUTINE  NAVIT(GC,NGA,NG,VA,ICA,CA,IXX,N6)                        NVIT0010
C   PGM=NXX. L.D.G.            VER.3  8-27-73      FORTRAN IV.     EBCD    NVIT0020
C   TO SET UP INITIAL POSITIONS AND PLANNED ROUTES                         NVIT0030
C   CALLED BY,   SETUPA                                                    NVIT0040
        DIMENSION  GC(8,6,NGA), VA(12), ICA(16), CA(16)                    NVIT0050
        DATA      RAD/0.0174533/,PI/3.14159/                               NVIT0060
 2000   FORMAT(1H0,5X,  'ERROR IN NAVIGATION INITIALIZATION PKG'/          NVIT0070
    1          6X,'CHECK TIMES IN TABLE OF GROUP CENTERS'/                 NVIT0080
    2          6X,'POINT=',I3,',  TIMES=',2F7.3,',    GROUP=',I3)          NVIT0090
C**  SETUP BORDERED TABLE  GC(8,6,NGA)                                     NVIT0100
 1      IXX = 0                                                            NVIT0110
        GC(1,1,NG) = 806.C                                                 NVIT0120
      DO   4 I=2,6
 4    GC(1,I,NG) = I
      GC(2,1,NG)=-10.
      GC(3,1,NG) = VA(10)
      GC(4,1,NG) = VA(7)
      GC(5,1,NG) = VA(4)
      GC(6,1,NG) =  VA(3)
      GC(7,1,NG) =    .25 + VA(3)
      GC(8,1,NG)=5.0+VA(3)
C** CENTER
 6    XC   = ICA(4)
      YC   = ICA(8)
      BR   = AINT(VA(1))
      RG   = (VA(1)-BR) * 1000.
C**  POINT A. RELFASE POINT
      X   = RG * SIN(BR*RAD)
      Y    = RG * COS(BR*RAD)
 8    GC(6,2,NG) = XC + X
      GC(6,3,NG) = YC + Y
      GC(6,4,NG) = VA(2)
      GC(6,6,NG) =  0.0
      XA   = XC + X
      YA   = YC + Y
      ZA   = VA(2)
      GC(6,5,NG) = VA(5)
C**  POINT B.
 10   BR   = AINT(VA(5))
      V    = (VA(5)-BR) * 1000.
      RG   = V * (VA(3)-VA(4))
      X    = RG * SIN(BR*RAD)
      Y    = RG * COS(BR*RAD)
      GC(5,2,NG) = XA - X
      GC(5,3,NG) = YA - Y
      GC(5,4,NG) = VA(6)
      GC(5,5,NG) = VA(5)
      GC(5,6,NG) = (VA(2)-VA(6)) / (VA(3)-VA(4))
C**  POINT A1.
 12   RG   = V * (GC(7,1,NG)-GC(6,1,NG) )
      GC(7,2,NG) = XA + RG*SIN(BR*RAD)
      GC(7,3,NG) = YA + RG*COS(BR*RAD)
      GC(7,4,NG) = ZA
      XR   = CA(12)
      YR   = CA(16)
      ZR   = ZA
 14   ANG = ATAN2( (XR-GC(7,2,NG)),(YR-GC(7,3,NG)) )
      IF (ANG.LT..0)     ANG = ANG + 2.* PI
      DANG = AINT(ANG/RAD)
      GC(7,5,NG) = DANG + V/1000.
      GC(7,6,NG) = 0.0
C**  POINT A2.
 18   RG   = V * (GC(8,1,NG) - GC(7,1,NG) )
      GC(8,2,NG) = GC(7,2,NG) + RG*SIN(DANG*RAD)
      GC(8,3,NG) = GC(7,3,NG) + RG*COS(DANG*RAD)
      GC(8,4,NG) = ZA
      GC(8,5,NG) = GC(7,5,NG)
      GC(8,6,NG) = 0.C                                                  NVIT0680
C**   PCINT C.                                                          NVIT0690
 20     BR   = AINT(VA(8))                                              NVIT0700
        V    = (VA(8)-BR) * 1000.                                       NVIT0710
        RG   = V * (VA(4)-VA(7) )                                       NVIT0720
        GC(4,2,NG) = GC(5,2,NG) - RG * SIN(BR*RAD)                      NVIT0730
        GC(4,3,NG) = GC(5,3,NG) - RG * COS(BR*RAD)                      NVIT0740
        GC(4,4,NG) = VA(9)                                              NVIT0750
        GC(4,5,NG) = VA(8)                                              NVIT0760
        GC(4,6,NG) = (VA(6)-VA(9)) / (VA(4)-VA(7))                      NVIT0770
C**   POINT D.                                                          NVIT0780
 22     BR   = AINT(VA(11))                                             NVIT0790
        V    = (VA(11)-BR) * 1000.                                      NVIT0800
        RG   = V * (VA(7)-VA(1C) )                                      VVIT0810
        GC(3,2,NG) = GC(4,2,NG) - RG * SIN(BR*RAD)                      NVIT0820
        GC(3,3,NG) = GC(4,3,NG) - RG * COS(BR*RAD)                      NVIT0830
        GC(3,4,NG) = VA(12)                                             NVIT0840
        GC(3,5,NG) = VA(11)                                             NVIT0850
        GC(3,6,NG) = (VA(9)-VA(12))/(VA(7)-VA(10))                      NVIT0860
C**   POINT E.                                                          NVIT0870
 24     XS   = CA(4)                                                    NVIT0880
        YS   = CA(8)                                                    NVIT0890
        ZS   = GC(3,4,NG)                                               NVIT0900
        ANG = ATAN2( (GC(3,2,NG)-XS),(GC(3,3,NG)-YS) )                  NVIT0910
        IF (ANG.LT..0)     ANG = ANG + 2.* PI                           NVIT0920
        DANG = AINT(ANG/RAD)                                            NVIT0930
 26     RG   = V * (GC(3,1,NG)-GC(2,1,NG) )                             NVIT0940
        XE   = GC(3,2,NG) - RG * SIN(DANG*RAD)                          NVIT0950
        YE   = GC(3,3,NG) - RG * COS(DANG*RAD)                          NVIT0960
        GC(2,2,NG) = XF                                                 NVIT0970
        GC(2,3,NG) = YE                                                 NVIT0980
        GC(2,4,NG) = GC(3,4,NG)                                         NVIT0990
        GC(2,5,NG) = DANG + V/1000.                                     NVIT1000
        GC(2,6,NG) = 0.0                                                NVIT1010
        IXX = 0                                                         NVIT1020
        DO 28 I=2,7                                                     NVIT1030
        IF (   GC(I+1,1,NG).GT.GC(I,1,NG) )      GO TO 28               NVIT1040
        IXX = IXX + 1                                                   NVIT1050
        WRITF(N6,2000) I, GC(I,1,NG), GC(I+1,1,NG), NG                  NVIT1060
 28     CONTINUF                                                        NVIT1070
        RETURN                                                          NVIT1080
        END                                                             NVIT1090


        SUBROUTINF  UNISET(KA1,VA1,LA,MA,LA1,MA1,NGMAX,KG,NUMAX,NBU,KBU, USET0010
     1               NAMBU,KBUK,BREL,BV,ID1,IXX)                         USET0020
C** PGM=NXX. L.D.G.          VER.3.  7-7-73     FORTRAN 1V      EBCD     USET0030
C MODS. N6=121.                                    VER.4   11-25-74 USET0040
C** TO SET UP UNITS FROM GROUP TABLES 1.1,1.2,2.1, ETC.                 USET0050
C     CALLED BY  SETUPA. AT SN = 78.                                    USET0060
        DIMENSION  KA1(LA,MA), VA1(LA,MA), NBU(NGMAX), KBUK(4,NUMAX)    USET0070
        DIMENSION  BREL(4,NUMAX),NAMBU(NUMAX,2),BV(NUMAX,8)             USET0080
 2005   FORMAT(1HO, 'ERROR IN UNISET AT E1=',F8.2,3X,'ID=',2I10)        USET0090
 2006   FORMAT(1HO, 'NUMBER OF UNITS EXCEEDS NUMAX =',I4,', AT ID=',I10) USET0100
```

```
2007  FORMAT(1HO,'IN GROUP TABLE=',I10,',  EXCLUDED LINE NC.=',I4)
1     IP2 = 0
      IXX = C
      N6  = 6
      KUG = 0
      DO 98 L=1,LA1
      KQ  = VA1(L,1) + .001
      IF (KA1(L,5).GT.C. AND .KQ.GT.0)  GO TO 86
85    F1  = 85.00
      WRITF(N6,2007) ID1, L
      GO TO 98
86    IF (KPU.LT.NUMAX)  GO TO 88
87    IXX = 1
      WRITF(N6,2006) NUMAX, ID1
      RFTURN
88    KBU = KBU + 1
      KUG = KUG + 1
      IF (KUG.FQ.1)   NBU(KG) = 100 * KBU
      KBUK(1,KBU) = ID1 + 100 * KUG
      KBUK(2,KBU) = 10000 * KA1(L,5)
      KBUK(3,KBU) = KG
      KBUK(4,KBU) = 1
      NAMBU(KBU,1) = KA1(L,2)
      NAMBU(KBU,2) = KA1(L,3)
      DO 90 LL=1,4
90    BREL(LL,KBU) = VA1(L,LL)
      DO 91 LL=1,8
91    BV(KBU,LL) = VA1(L,LL+4)
      BV(KBU,3) = BV(KBU,3) / 6.080
      BV(KBU,8) = BV(KBU,8) / 6.080
C**  TAKE CARE OF QUANTITY.GT.1
92    IF (KQ.LF.1)  GO TO 98
      IF (KBU.LT.NUMAX)  GO TO 94
      GO TO 87
94    KQ  = KQ - 1
      K   = KBU
      KBU = KBU + 1
      KUG = KUG + 1
      KBUK(1,KBU) = KBUK(1,K) + 100
      KBUK(2,KBU) = KBUK(2,K)
      KBUK(3,KBU) = KG
      KBUK(4,KBU) = 1
      NAMBU(KBU,1) = NAMBU(K,1)
      NAMBU(KBU,2) = NAMBU(K,2)
      DO 96 LL=1,4
96    BREL(LL,KBU) = BREL(LL,K)
      DO 97 LL=1,8
97    BV(KBU,LL) = BV(K,LL)
      GO TO 92
98    CONTINUF
      NBU(KG) = NBU(KG) + KUG
      RETURN
      END
      SUBROUTINE  SETUPB(IPRINT)                                        SETB0010
C  PGM=NXX.  L.D.GREGORY.  10-1-73        FORTRAN IV              EBCD  SETB0020
C  TO SET UP PLATFORMS VS SYSTEMS FOR CURRENT PROBLEM                   SETB0030
C  CALLED BY MAIN AND ZIP CARD AS FOLLOWS                               SETB0040
CIP  8 2 P                               SET UP SYSTEM AND SUBSYSTEMS   SETB0050
C**                                                                     SETB0060
     COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              SETB0070
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                      SETB0080
     1,IDUMB(72), NFLAG, NFLAG2                                         SETB0090
CZAZA                                                                   SETB0100
      INTEGER TITLA                                                     SETB0110
      COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)     SETB0120
     1, TITLA(15),HEADA(37),IFMT                                        SETB0130
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                          SETB0140
      DIMENSION  BZA( 555)                                              SETB0150
      EQUIVALENCE  (NCBA,BZA(1))                                        SETB0160
      DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/                            SETB0170
CNAVIG                                                                  SETB0180
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                  SETB0190
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),      SETB0200
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),      SETB0210
     3          TTIME,NUMX,                                             SETB0220
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),   SETB0230
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)    SETB0240
      COMMON/CWORK/LA2,MA2,LAA2,MAA2, KA2(3,15),VA2(12,15),             SETB0250
     1              LA3,MA3,LAA3,MAA3, VA3(48,15),          KA5( 547)   SETB0260
CPLAT   VS   SYSTEMS  (SUBSYSTEMS)                                      SETB0270
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),           SETB0280
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),         SETB0290
     2  BPX1(12,15),                                                    SETB0300
     3              NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),          SETB0310
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),         SETB0320
     5  RPX1(12,15)                                                     SETB0330
C***                                                                    SETB0340
C  NAMBP  = NAME,BLU PLATFORMS          / NAMRP  = NAME,RED PLATFORMS   SETB0350
C  KBPT   = KODES,BLU PLAT TYPE,61130000/ KRPT   = ETC.                 SETB0360
C  NABSS  = NAME,BLU SUB SYSTEMS        / ETC.   =                      SETB0370
C  KBST   = KODES,BLU SS TYPES, 65140000/        =                      SETB0380
C  KBPXS  = KODES,BLU PLAT VS SS + QTY  /        =                      SETB0390
C         = E.G. 65140002               /        =                      SETB0400
C  NBP    = NO.OF BLU PLAT IN  KBPT     /        =                      SETB0410
C  NBSS   = NO.OF BLU SUBSYS  IN KBST   /        =                      SETB0420
C  NBSPP  = NO.SYS PER PLATFORM                                         SETB0430
C  EPX1   = BLU PLAT EXTENT(-1) DATA    / RPX1   = RED PLAT EXTENT(-1)  SETB0440
      DIMENSION  KBPTT( 7), KRPTT( 3)                                   SETB0450
      CATA  NBPTT,KBPTT/ 7,61100000,61200000,61300000,61500000,         SETB0460
     1   62100003,631C0000,63200000/                                    SETB0470
      DATA  NRPTT,KRPTT/ 3,81300000,82100003,83300000/                  SETB0480
      CATA  BLANK,LANK/'    ','    '/                                   SETB0490
 2005 FORMAT(1H0,22HERROR IN SETUPB AT E1=,F8.2,3X,'ID1=',2I10)         SETB0500
 2006 FORMAT(1H0, 'NO. BLU PLATFORMS=',I4,',  EXCEEDS STORAGE=',I4)     SETB0510
 2007 FORMAT(1H0, 'NO TABLE FOUND FOR ID1, ID2=',2I10)                  SETB0520
 2008 FORMAT(1H0, 'NO. BLU SYSTEMS=',I4,',  EXCEEDS STORAGE=',I4)       SETB0530
 2010 FORMAT(6X,'BLU PLATFORM DATA')                                    SETB0540
 2012 FORMAT(6X,'PLATFORM TYPE=',I10/(9X,6G14.6) )                      SETB0550
 2206 FORMAT(1H0, 'NO. RED PLATFORMS=',I4,',  EXCEEDS STORAGE=',I4)
 2208 FORMAT(1H0, 'NO. RED SYSTEMS=',I4,',  EXCEEDS STORAGE=',I4)
 2210 FORMAT(6X,'RED PLATFORM DATA')
      LAA5   = 727
      LAA3   = 48
      MAA3   = 15
      LAA2   = 12
      MAA2   = 15
      NBPMX  = 15
      NBSSMX= 45
      NRPMX  = 15
      NRSSMX= 45
      KZARL  = 96 + LA*(5 + MA)
      JZARL  = KZARL
      NFLAG2= 0
C***  FIND DISTINCT BLU PLATFORMS (UNIT) CODES
 1    IF (KBU.LE.0)  GO TO 201
      DO 12 I=1,KBU
 12   KA5(I)= KBUK(2,I)
      CALL  KORDER(KA5,KBU)
      CALL  UNIQUE(KA5,KBU,NBP)
      IF (NBP.LE.NBPMX)  GO TO 14
      WRITE(N6,2006) NBP, NBPMX
      NFLAG2 = NFLAG2 + 1
      NBP    = NBPMX
 14   DO 16 I=1,NBP
 16   KBPT(I) = KA5(I)
C***  ZERO OUT WORKING ARRAYS
      DO 22 J=1,NBPMX
      DO 18 I=1,2
      NAMBP(I,J) = LANK
 18   KA2(I,J)   =  LANK
      KA2(3,J)   =  0
      DO 20 I=1,12
      VA2(I,J) = .0
 20   VA3(I,J) = .0
      DO 22 I=13,LAA3
 22   VA3(I,J) = .0
C***  CYCLE THRU CATALOG PLATFORM TABLES
      IY  = 3
      DO 44  KC=1,NBPTT
      ID1 =(KBPTT(KC) / 10000) * 10000
      DO 42  KX=1,5
      ID2 = -KX
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX.EQ.0)  GO TO 26
      KOUTA = MOD(KBPTT(KC),10000)
      DO 24 I=1,4
      KOUT = MOD(KOUTA,10)
      IF (KX.EQ.KOUT)  GO TO 42
 24   KOUTA = KOUTA / 10
      WRITE(N6,2007) ID1,ID2
      GO TO 42
C***  CYCLE THRU PLATFORM TYPES IN KBPT (FROM ENGAGEMENT STRUCTURE)
 26   DO 40  KT=1,NBP
```

```
      K    =  KBPT(KT) / 10000                                          SETB1110
      DO 28  J=1,LA1                                                    SETB1120
      IF (K.EQ.KA1( J,5))  GO TO 30                                     SETB1130
 28   CONTINUE                                                          SETB1140
      GO TO 40                                                          SETB1150
C***  FOUND, PLATFORM TYPE IN CATALOG                                   SETB1160
 30   JT   = J                                                          SETB1170
      IF (KX.GE.2)  GO TO 36                                            SETB1180
C***  TRANSFER PLATFORM NAME, TYPE, AND DATA                            SETB1190
 32   KA2(1,KT) = KA1(JT,2)                                             SETB1200
      KA2(2,KT) = KA1(JT,3)                                             SETB1210
      KA2(3,KT) = KBPT(KT)                                              SETB1220
      NAMBP(1,KT) = KA2(1,KT)                                           SETB1230
      NAMBP(2,KT) = KA2(2,KT)                                           SETB1240
      DO 34 J=1,12                                                      SETB1250
 34   VA2(J,KT) = VA1(JT,J)                                             SETB1260
      GO TO 40                                                          SETB1270
 36   KTEST = 10000 * K                                                 SETB1280
      DO 37 J=1,NBP                                                     SETB1290
      IF (KTEST.EQ.KA2(3,J) )  GO TO 371                                SETB1300
 37   CONTINUE                                                          SETB1310
      GO TO 40                                                          SETB1320
 371  JA   = 12*(KX-2)                                                  SETB1330
      DO 38 J=1,12                                                      SETB1340
 38   VA3(JA+J,KT) = VA1(JT,J)                                          SETB1350
 40   CONTINUE                                                          SETB1360
 42   CONTINUE                                                          SETB1370
 44   CONTINUE                                                          SETB1380
C***  BUILD SYSTEM LIST (UNIQUE)                                        SETB1390
 50   LA5 = 0                                                           SETB1400
      DO 52 J=1,NBP                                                     SETB1410
      DO 52 I=1,LAA3                                                    SETB1420
      A    = VA3(I,J)                                                   SETB1430
      IF (A.LE..001)  GO TO 52                                          SETB1440
      IF ( MOD( INT((A+.004)*100.), 100).LT.1)   GO TO 52               SETB1450
      LA5 = LA5 + 1                                                     SETB1460
      KA5(LA5) = (A + .004)                                             SETB1470
 52   CONTINUE                                                          SETB1480
      CALL  KORDER(KA5,LA5)                                             SETB1490
      CALL  UNIQUE(KA5,LA5,K)                                           SETB1500
      NBSS = K                                                          SETB1510
      IF (NBSS.LE.NBSSMX)  GO TO 54                                     SETB1520
      WRITE(N6,2008) NBSS, NBSSMX                                       SETB1530
      NFLAG2 = NFLAG2 + 1                                               SETB1540
      NBSS = NBSSMX                                                     SETB1550
 54   DO 56 I=1,NBSS                                                    SETB1560
 56   KBST(I) = 10000*KA5(I)                                            SETB1570
C***  BUILD SYSTEMS VS PLATFORMS MATRIX  KBPXS                          SETB1580
      DO 60 I=1,NBP                                                     SETB1590
 60   NBSPP(I) = 0                                                      SETB1600
      DO 66 I=1,NBSS                                                    SETB1610
      NABSS(1,I) = LANK                                                 SETB1620
      NABSS(2,I) = LANK                                                 SETB1630
      NBPWS(I) = 0                                                      SETB1640
      KST = KBST(I)                                                     SETB1650
      DO 66 J=1,NBP                                                     SETB1660
      KBPXS(I,J) = 0                                                    SETB1670
      DO 64 K=1,LAA3                                                    SETB1680
      A    = VA3(K,J)                                                   SETB1690
      IF (A.LE..001)  GO TO 64                                          SETB1700
      L    = INT(A + .004)*10000                                        SETB1710
      IF (L.NE.KST)  GO TO 64                                           SETB1720
      KL   = MOD( INT((A+.004)*100.), 100)                              SETB1730
      IF (K.GT.36)  GO TO 62                                            SETB1740
      KBPXS(I,J) = KST + MIN0(KL,99)                                    SETB1750
      NBPWS(I) = NBPWS(I) + 1                                           SETB1760
      NBSPP(J) = NBSPP(J) + 1                                           SETB1770
      GO TO 64                                                          SETB1780
 62   KBPXS(I,J) = KBPXS(I,J) + 100*MIN0(KL,99)                         SETB1790
 64   CONTINUE                                                          SETB1800
 66   CONTINUE                                                          SETB1810
C***  TRANSFER EXTENT(-1) DATA                                           SETB1820
 74   DO 76 I=1,LAA2                                                    SETB1830
      DO 76 J=1,MAA2                                                    SETB1840
 76    BPX1(I,J) = VA2(I,J)                                             SETB1850
C***  PRINT OPTION                                                       SETB1860
      IF (IPRINT.LT.2)  GO TO 78                                        SETB1870
      CALL PAGE                                                         SETB1880
      WRITE(N6,2010)                                                    SETB1890
      DO 77 I=1,NBP                                                     SETB1900
 77   WRITE(N6,2012)  KBPT(I),(BPX1(J,I),J=1,12)                        SETB1910
 78   CONTINUE                                                          SETB1920
C***  PROCESS RED PLATFORM DATA                                           SETB1930
C***  FIND DISTINCT RED PLATFORMS (UNIT) CODES                            SETB1940
 201   IF (KRU.LE.0)  GO TO 401                                         SETB1950
      DO 212 I=1,KRU                                                    SETB1960
 212   KA5(I)= KRUK(2,I)                                                SETB1970
       CALL  KORDER(KA5,KRU)                                            SETB1980
       CALL  UNIQUE(KA5,KRU,NRP)                                        SETB1990
       IF (NRP.LE.NRPMX)  GO TO 214                                     SETB2000
       WRITE(N6,2206) NRP, NRPMX                                        SETB2010
       NFLAG2 = NFLAG2 + 1                                              SETB2020
       NRP    = NRPMX                                                   SETB2030
 214  DO 216 I=1,NRP                                                    SETB2040
 216   KRPT(I) = KA5(I)                                                 SETB2050
C***  ZERO OUT WORKING ARRAYS                                            SETB2060
      DO 222 J=1,NRPMX                                                  SETB2070
      DO 218 I=1,2                                                      SETB2080
       NAMRP(I,J) = LANK                                                SETB2090
 218   KA2(I,J)  =  LANK                                                SETB2100
       KA2(3,J)  =  0                                                   SETB2110
      DO 220 I=1,12                                                     SETB2120
       VA2(I,J) = .0                                                    SETB2130
 220   VA3(I,J) = .0                                                    SETB2140
      DO 222 I=13,LAA3                                                  SETB2150
 222   VA3(I,J) = .0                                                    SETB2160
C***  CYCLE THRU CATALOG PLATFORM TABLES                                 SETB2170
      IY  = 3                                                           SETB2180
      DO 244 KC=1,NRPTT                                                 SETB2190
      ID1 =(KRPTT(KC) / 10000) * 10000                                  SETB2200
      DO 242 KX=1,5                                                     SETB2210
      ID2 = -KX                                                         SETB2220
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                  SETB2230
      IF (IXX.EQ.0)  GO TO 226                                          SETB2240
      KOUTA = MOD(KRPTT(KC),10000)                                      SETB2250
      DO 224 I=1,4                                                      SETB2260
      KOUT = MOD(KOUTA,10)                                              SETB2270
      IF (KX.EQ.KOUT)  GO TO 242                                        SETB2280
  224 KOUTA = KOUTA / 10                                                SETB2290
      WRITE(N6,2007) ID1,ID2                                            SETB2300
      GO TO 242                                                         SETB2310
C***  CYCLE THRU PLATFORM TYPES IN KRPT (FROM ENGAGEMENT STRUCTURE)      SETB2320
  226 DO 240 KT=1,NRP                                                   SETB2330
      K    =  KRPT(KT) / 10000                                          SETB2340
      DO 228 J=1,LA1                                                    SETB2350
      IF (K.EQ.KA1( J,5))  GO TO 230                                    SETB2360
  228 CONTINUE                                                          SETB2370
      GO TO 240                                                         SETB2380
C***  FOUND, PLATFORM TYPE IN CATALOG                                   SETB2390
  230 JT  = J                                                           SETB2400
      IF (KX.GE.2)  GO TO 236                                           SETB2410
C***  TRANSFER PLATFORM NAME, TYPE, AND DATA                            SETB2420
  232 KA2(1,KT) = KA1(JT,2)                                             SETB2430
      KA2(2,KT) = KA1(JT,3)                                             SETB2440
      KA2(3,KT) = KRPT(KT)                                              SETB2450
      NAMRP(1,KT) = KA2(1,KT)                                           SETB2460
      NAMRP(2,KT) = KA2(2,KT)                                           SETB2470
     DO 234 J=1,12                                                      SETB2480
  234 VA2(J,KT) = VA1(JT,J)                                             SETB2490
      GO TO 240                                                         SETB2500
  236 KTEST = 10000 * K                                                 SETB2510
     DO 237 J=1,NRP                                                     SETB2520
      IF (KTEST.EQ.KA2(3,J) )  GO TO 2371                               SETB2530
  237 CONTINUE                                                          SETB2540
      GO TO 240                                                         SETB2550
 2371 JA  = 12*(KX-2)                                                   SETB2560
     DO 238 J=1,12                                                      SETB2570
  238 VA3(JA+J,KT) = VA1(JT,J)                                          SETB2580
  240 CONTINUE                                                          SETB2590
  242 CONTINUE                                                          SETB2600
  244 CONTINUE                                                          SETB2610
C***  BUILD SYSTEM LIST (UNIQUE)                                        SETB2620
  250 LA5 = 0                                                           SETB2630
     DO 252 J=1,NRP                                                     SETB2640
     DO 252 I=1,LAA3                                                    SETB2650
      A   = VA3(I,J)                                                    SETB2660
      IF (A.LE..001)  GO TO 252                                         SETB2670
      IF ( MOD( INT((A+.004)*100.), 100).LT.1)   GO TO 252              SETB2680
      LA5 = LA5 + 1                                                     SETB2690
      KA5(LA5) = (A + .004)                                             SETB2700
  252 CONTINUE                                                          SETB2710
      CALL  KORDER(KA5,LA5)                                             SETB2720
      CALL  UNIQUE(KA5,LA5,K)                                           SETB2730
      NRSS = K                                                          SETB2740
      IF (NRSS.LE.NRSSMX)  GO TO 254                                    SETB2750
      WRITE(N6,2208) NRSS, NRSSMX                                       SETB2760
      NFLAG2 = NFLAG2 + 1                                               SETB2770
      NRSS = NRSSMX                                                     SETB2780
  254 DO 256 I=1,NRSS                                                   SETB2790
  256  KRST(I) = 10000*KA5(I)                                           SETB2800
C***  BUILD SYSTEMS VS PLATFORMS MATRIX  KRPXS                          SETB2810
      DO 260 I=1,NRP                                                    SETB2820
  260  NRSPP(I) = 0                                                     SETB2830
      DO 266 I=1,NRSS                                                   SETB2840
       NARSS(1,I) = LANK                                                SETB2850
       NARSS(2,I) = LANK                                                SETB2860
       NRPWS(I) = 0                                                     SETB2870
       KST = KRST(I)                                                    SETB2880
      DO 266 J=1,NRP                                                    SETB2890
       KRPXS(I,J) = 0                                                   SETB2900
      DO 264 K=1,LAA3                                                   SETB2910
       A    = VA3(K,J)                                                  SETB2920
       IF (A.LE..001)  GO TO 264                                        SETB2930
       L    = INT(A + .004)*10000                                       SETB2940
       IF (L.NE.KST)  GO TO 264                                         SETB2950
       KL   = MOD( INT((A+.004)*100.), 100)                             SETB2960
       IF (K.GT.36)  GO TO 262                                          SETB2970
       KRPXS(I,J) = KST + MINO(KL,99)                                   SETB2980
       NRPWS(I) = NRPWS(I) + 1                                          SETB2990
       NRSPP(J) = NRSPP(J) + 1                                          SETB3000
       GO TO 264                                                        SETB3010
  262  KRPXS(I,J) = KRPXS(I,J) + 100*MINO(KL,99)                        SETB3020
  264  CONTINUE                                                         SETB3030
  266  CONTINUE                                                         SETB3040
C***  TRANSFER EXTENT(-1) DATA                                          SETB3050
  274 DO 276 I=1,LAA2                                                   SETB3060
      DO 276 J=1,MAA2                                                   SETB3070
  276   RPX1(I,J) = VA2(I,J)                                            SETB3080
C***  PRINT OPTION                                                      SETB3090
       IF (IPRINT.LT.2)  GO TO 278                                      SETB3100
       CALL PAGE                                                        SETB3110
       WRITE(N6,2210)                                                   SETB3120
       DO 277 I=1,NRP                                                   SETB3130
  277  WRITE(N6,2012)  KRPT(I),(RPX1(J,I),J=1,12)                       SETB3140
  278  CONTINUE                                                         SETB3150
  401  RETURN                                                           SETB3160
       END                                                              SETB3170


       SUBROUTINE  SETUPC(IPRINT)                                       SETC0010
C  PGM=NXX(NEM). L.D.G.        VER.2     10-22-73   FORTRAN IV    EBCD  SETC0020
C  TO SET UP SYSTEMS DATA FOR CURRENT PROBLEM FOR SYSTEMS IN SETUPB     SETC0030
C  SETS UP,BLU 1.ASMS=7800    2.TRAJ=4110                               SETC0040
C                                                                       SETC0050
C           ,REC 1.ASMS=9800    2.TRAJ=5110    3.SSMS=9600              SETC0060
C  CALLED BY MAIN AND ZIP CARD AS FOLLOWS                               SETC0070
CIP  8 3 P                                 SET UP SYSTEM DATA           SETC0080
C***                                                                    SETC0090
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             SETC0100
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                        SETC0110
     1,IDUMB(72), NFLAG, NFLAG2                                           SETC0120
CZAZA                                                                     SETC0130
      INTEGER TITLA                                                       SETC0140
      COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)       SETC0150
     1, TITLA(15),HEADA(37),IFMT                                          SETC0160
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                            SETC0170
      DIMENSION  BZA( 555)                                                SETC0180
      EQUIVALENCE  (NCBA,BZA(1))                                          SETC0190
      DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/                              SETC0200
CNAVIG                                                                    SETC0210
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                    SETC0220
     1    NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),         SETC0230
     2    NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),         SETC0240
     3        TTIME,NUMX,                                                 SETC0250
     4    KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),      SETC0260
     5    KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)       SETC0270
CPLAT   VS  SYSTEMS  (SUBSYSTEMS)                                         SETC0280
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),             SETC0290
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),           SETC0300
     2  BPX1(12,15),                                                      SETC0310
     3               NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),           SETC0320
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),           SETC0330
     5  RPX1(12,15)                                                       SETC0340
CMSLSYS                        16      2      2      6      6     24    50 SETC0350
      COMMON/CMSLSY/ NMSL,NMSLMX,NBSSM,NBASM,NRSSM,NRASM,LATMSL,LTRAJ,    SETC0360
     1   NAMMSL(2,16),KMSLTY(16),                                         SETC0370
     2   DATMSL(24,16), TTRAJ(50,16)                                      SETC0380
CWORK                                                                     SETC0390
      COMMON/CWORK/                                                       SETC0400
     1   LW1,MW1,LAW1,MAW1,  KW1(27,5),  VW1(27,12),                      SETC0410
     2   LW2,MW2,LAW2,MAW2,  KW2(27,5),  VW2(27,12),                      SETC0420
     3   LW3,MW3,LAW3,MAW3,  KW3(27,5),  VW3(27,12),          PADW(111)   SETC0430
      DATA  BLANK,LANK/ '    ','    '/                                    SETC0440
CFORMATS                                                                  SETC0450
 2006 FORMAT(1H0,'NO.OF SSM+ASM=',I4,',  EXCEEDS STORAGE=',I4)            SETC0460
 2007 FORMAT(1H0,'NO TABLE FOUND FOR ID1, ID2=', 2I10)                    SETC0470
 2008 FORMAT(1H0,'NO LINE FOUND IN TABLE ID1, ID2=',I9,',',I4,            SETC0480
    1 ',  FOR MISSILE TYPE=',I9)                                          SETC0490
 2009 FORMAT(1H0,'NO LINE FOUND IN TABLE ID1, ID2=',I9,',',I4,            SETC0500
    1 ',  FOR MISSILE TYPE=',I9,',  TRAJ TYPE=',I9)                       SETC0510
 2010 FORMAT(1H0,'ERROR IN TABLE ID1, ID2=',I9,',',I4,                    SETC0520
    1 ',  FOR TRAJ=',I9)                                                  SETC0530
 2011 FORMAT(1H0,'AT END OF SETUPC, CUM NO. SETUP ERRORS=NFLAG=',I5)      SETC0540
 3020 FORMAT(6X,'SSMS & ASMS'/                                            SETC0550
     1   8X,'NO.BLU SSM TYPES=',I4,',  NO.BLU ASM TYPES=',I4/             SETC0560
     2   8X,'NO.RED SSM TYPES=',I4,',  NO.RED ASM TYPES=',I4)             SETC0570
 3022 FORMAT(/1X,I6,'.',2A4,I9,'    MISSILE DATA')                        SETC0580
 3023 FORMAT(/1X,I6,'.',2A4,I9,'    TEMP TRAJ DATA')                      SETC0590
 3024 FORMAT( 8X,10G12.5)                                                 SETC0600
CFORMATS                                                                  SETC0610
      KZARL = 96 + LA*(5 + MA)                                            SETC0620
      JZARL = KZARL                                                       SETC0630
      NFLAG2= 0                                                           SETC0640
C***  INITIALIZE                                                          SETC0650
```

```
 200    NMSLMX = 16
        LATMSL = 24
        LTRAJ  = 50
        NMSL   = 0
        NBSSM  = 0
        NBASM  = 0
        NRSSM  = 0
        NRASM  = 0
        DO 203 J=1,NMSLMX
        DO 201 I=1,2
 201    NAMMSL(I,J) = LANK
        KMSLTY(J) = 0
        DO 202 I=1,LATMSL
        DATMSL(I,J) = .0
 202    TTRAJ (I,J) = .0
        DO 203 I=LATMSL,LTRAJ
 203    TTRAJ (I,J) = .0
        KEX  = 0
         IF (KBU.LE.0)  GO TO 300
C***  SET UP BLU.   KEX = 1
 204    KEX = 1
        LW1 = 0
        LW2 = 0
        LW3 = 0
C**CALL RAW TRAJ
        IY   = 3
        KTRAN = 1
         ID1    = 41100000
         ID2    = -1
C*** READ FROM DISK 11
 206    CALL  RD3D11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
        IF ( IXX.EQ.0)      GO TO (210,218,228,236), KTRAN
        WRITE(N6,2007)  ID1,ID2
        NFLAG2 = NFLAG2 + 1
        GO TO (216,224,232,300), KTRAN
C**TRANSFER RAW TRAJ TO CWORK (1)
 210    DO 213 I=1,LA1
        DO 212 J=1,MA1
 212    VW1(I,J) = VA1(I,J)
 213    KW1(I,5) = KA1(I,5)
        LW1 = LA1
        MW1 = MA1
C**CALL BLU MSL (-2)
 216    KTRAN = 2
         ID1    = 78100000
         ID2    = -2
        GO TO 206
C**TRANSFER MSL (-2) TO CWORK (2)
 218    DO 222 I=1,LA1
        DO 220 J=1,MA1
 220    VW2(I,J) = VA1(I,J)
 222    KW2(I,5) = KA1(I,5)
        LW2 = LA1
        MW2 = MA1
C**CALL BLU MSL (-3)
```

```
 224  KTRAN = 3
      ID1    = 78100000
      ID2    = -3
C**   GO TO 206          USE THIS INSTEAD OF 232 IF EXTENT -3 IS ADDED
      GO TO 232
C**TRANSFER MSL (-2) TO CWORK (3)
 228  DO 230 I=1,LA1
      KW3(I,5) = KA1(I,5)
      DO 230 J=1,MA1
 230  VW3(I,J) = VA1(I,J)
      LW3 = LA1
      MW3 = MA1
C**CALL BLU MSL (-1)
 232  KTRAN = 4
      ID1    = 78100000
      ID2    = -1
      GO TO 206
C*** PROCESS BLU MSL NOW IN CZAZA AND CWORK
C**TEST IF ALL NEEDED TABLES WERE FOUND. INCLUDE LW3 IF ADDED
 236  IF (LW1*LW2    .EQ.0)  GO TO 300
C**CYCLE THRU SYSTEMS IN KBST
      DO 280 KT=1,NBSS
      KTYPE = KBST(KT)
      NFLAG3 = 0
C**CYCLE THRU 7810 (-1) IN CZAZA
      KTEST = KTYPE / 10000
      DO 240 I=1,LA1
      IF (KTEST.EQ.KA1(I,5) )   GO TO 242
 240  CONTINUE
      GO TO 280
C**TEST IF STORAGE.EXCEEDED
 242  KL   = I
      NMSL = NMSL + 1
      IF (NMSL.LE.NMSLMX)  GO TO 243
      WRITE(N6,2006)  NMSL, NMSLMX
      NFLAG2 = NFLAG2 + 1
      GO TO 400
C**SET COUNTERS & NAME
 243  KTEST = KTYPE / 1000000
      IF (KTEST.EQ.76)  NBSSM = NBSSM + 1
      IF (KTEST.EQ.78)  NBASM = NBASM + 1
      DO 244 I=1,2
      NABSS(I,KT) = KA1(KL,I+1)
 244  NAMMSL(I,NMSL) = NABSS(I,KT)
C**STORE TYPE & DATA FROM EXTENT (-1)
      KMSLTY(NMSL) = KTYPE
      DO 246 I=1,6
 246  CATMSL(I+15,NMSL) = VA1(KL,I)
C**STORE DATA FROM EXTENT (-2) IN CWORK (2)
 250  KTEST = KTYPE / 10000
      DO 252 I=1,LW2
      IF (KTEST.EQ.KW2(I,5) )  GO TO 254
 252  CONTINUE
      NFLAG3 = NFLAG3 + 1
      ID1 = 78100000
      ID2 = -2
      WRITE(N6,2008)  ID1, ID2, KTYPE
      GO TO 260
254   KL   = I
      DO 256 I=1,MW2
256   DATMSL(I,NMSL) = VW2(KL,I)
C**STORE DATA FROM EXTENT (-3) IN CWORK (3)         (IF ANY)
260   IF (LW3.EQ.0.OR.LATMSL.LT.36)  GO TO 270
      KTEST = KTYPE / 10000
      DO 262 I=1,LW3
      IF (KTEST.EQ.KW3(I,5) )  GO TO 264
262   CONTINUE
      NFLAG3 = NFLAG3 + 1
      ID1 = 78100000
      ID2 = -3
      WRITE(N6,2008)  ID1, ID2, KTYPE
      GO TO 270
264   KL   = I
      DO 266 I=1,MW3
266   DATMSL(I+24,NMSL) = VW3(KL,I)
C***  PROCESS TEMP TRAJ & STORE IN  TTRAJ
270   ID1 = 41100000
      ID2 = -1
      KTRTY = MOD(IFIX(DATMSL(2,NMSL)*100.+.4),100)
      DO 276 K=1,2
      KTEST  = KTRTY*100 + K
      DO 272 I=1,LW1
      IF (KTEST.EQ.KW1(I,5) )  GO TO 274
272   CONTINUE
      IF (K.EQ.2)  GO TO 278
      NFLAG3 = NFLAG3 + 1
      ID 1 = 41100000
      ID2 = -1
      WRITE(N6,2009) ID1,ID2,KTYPE, KTEST
      GO TO 278
274   KL   = I
      CALL   STTRAJ(K,KTEST,KL,VW1,LA,MA,IXX,TTRAJ(1,NMSL),LTRAJ)
      IF (IXX.EQ.0)  GO TO 276
      NFLAG3 = NFLAG3 + 1
      WRITE(N6,2010) ID1,ID2, KTEST
276   CONTINUE
C*** TEST FOR ERROR FREE SETUP.  PUT POINTER IN SYSTEM LIST.
278   IF (NFLAG3.LE.0)   KBST(KT) = KBST(KT) + NMSL
280   CONTINUE
C***
C***      ***       ***       ***
C***SET UP RED
300   IF (KRU.LE.0)  GO TO 381
304   KFX = 2
      LW1 = 0
      LW2 = 0
      LW3 = 0
      IY  = 3
      KTRAN = 1
      ID1   = 51100000
      ID2    = -1
 306  CALL   RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX.EQ.0)        GO TO (310,318,328,336), KTRAN
      WRITE(N6,2007)   ID1,ID2
      NFLAG2 = NFLAG2 + 1
      GO TO (316,324,332,400), KTRAN
 310  DO 313 I=1,LA1
      DO 312 J=1,MA1
 312  VW1(I,J) = VA1(I,J)
 313  KW1(I,5) = KA1(I,5)
      LW1 = LA1
      MW1 = MA1
 316  KTRAN = 2
      ID1    = 98100000
      ID2    = -2
      GO TO 306
C**TRANSFER MSL (-2) TO CWORK (2)
 318  DO 322 I=1,LA1
      DO 320 J=1,MA1
 320  VW2(I,J) = VA1(I,J)
 322  KW2(I,5) = KA1(I,5)
      LW2 = LA1
      MW2 = MA1
C**CALL RED MSL (-3)
 324  KTRAN = 3
      ID1    = 98100000
      ID2    = -3
C**   GO TO 306             USE THIS INSTEAD OF 332 IF EXTENT -3 IS ADDED
      GO TO 332
C**TRANSFER MSL (-3) TO CWORK (3)
 328  DO 330 I=1,LA1
      KW3(I,5) = KA1(I,5)
      DO 330 J=1,MA1
 330  VW3(I,J) = VA1(I,J)
      LW3 = LA1
      MW3 = MA1
C**CALL RED MSL (-1)
 332  KTRAN = 4
      ID1    = 98100000
      ID2    = -1
      GO TO 306
C*** PROCESS RED MSL NOW IN CZAZA AND CWORK
C**TEST IF ALL NEEDED TABLES WERE FOUND. INCLUDE LW3 IF ADDED
 336  IF (LW1*LW2    .EQ.0)  GO TO 400
C**CYCLE THRU SYSTEMS IN KRST
      DO 380 KT=1,NRSS
      KTYPE = KRST(KT)
      NFLAG3 = 0
C**CYCLE THRU 9810 (-1) IN CZAZA
      KTEST = KTYPE / 10000
      DO 340 I=1,LA1
      IF (KTEST.EQ.KA1(I,5) )   GO TO 342
 340  CONTINUE
      GO TO 380
C**TEST IF STORAGE EXCEEDED
```

```
 342    KL   = I
        NMSL = NMSL + 1
        IF (NMSL.LE.NMSLMX)  GO TO 343
        WRITE(N6,2006)  NMSL, NMSLMX
        NFLAG2 = NFLAG2 + 1
        GO TO 400
C**SET COUNTERS & NAME
 343    KTEST = KTYPE / 1000000
        IF (KTEST.EQ.96)  NRSSM = NRSSM + 1
        IF (KTEST.EQ.98)  NRASM = NRASM + 1
        DO 344 I=1,2
        NARSS(I,KT) = KA1(KL,I+1)
 344    NAMMSL(I,NMSL) = NARSS(I,KT)
C**STORE TYPE & DATA FROM EXTENT (-1)
        KMSLTY(NMSL) = KTYPE
        DO 346 I=1,6
 346    DATMSL(I+15,NMSL) = VA1(KL,I)
C**STORE DATA FROM EXTENT (-2) IN CWORK (2)
 350    KTEST = KTYPE / 10000
        DO 352 I=1,LW2
        IF (KTEST.EQ.KW2(I,5) )  GO TO 354
 352    CONTINUE
        NFLAG3 = NFLAG3 + 1
        ID1 = 98100000
        ID2 = -2
        WRITE(N6,2008)  ID1, ID2, KTYPE
        GO TO 360
 354    KL   = I
        DO 356 I=1,MW2
 356    DATMSL(I,NMSL) = VW2(KL,I)
C**STORE DATA FROM EXTENT (-3) IN CWORK (3)       (IF ANY)
 360    IF (LW3.EQ.0.OR.LATMSL.LT.36)  GO TO 370
        KTEST = KTYPE / 10000
        DO 362 I=1,LW3
        IF (KTEST.EQ.KW3(I,5) )  GO TO 364
 362    CONTINUE
        NFLAG3 = NFLAG3 + 1
        ID1 = 98100000
        ID2 = -3
        WRITE(N6,2008)  ID1, ID2, KTYPE
        GO TO 370
 364    KL   = I
        DO 366 I=1,MW3
 366    DATMSL(I+24,NMSL) = VW3(KL,I)
C***  PROCESS TEMP TRAJ & STORE IN TTRAJ
 370    ID1 = 51100000
        ID2 = -1
        KTRTY = MOD(IFIX(DATMSL(2,NMSL)*100.+.4),100)
        DO 376 K=1,2
        KTEST = KTRTY*100 + K
        DO 372 I=1,LW1
        IF (KTEST.EQ.KW1(I,5) )  GO TO 374
 372    CONTINUE
        IF (K.EQ.2)  GO TO 378
        NFLAG3 = NFLAG3 + 1
```

```
      WRITE(N6,2009) ID1,ID2,KTYPE,KTEST                                SETC3410
      GO TO 378                                                         SETC3420
  374 KL   = I                                                          SETC3430
      CALL   STTRAJ(K,KTEST,KL,VW1,LA,MA,IXX,TTRAJ(1,NMSL),LTRAJ)       SETC3440
      IF (IXX.EQ.0)  GO TO 376                                          SETC3450
      NFLAG3 = NFLAG3 + 1                                               SETC3460
      WRITE(N6,2010) ID1,ID2, KTEST                                     SETC3470
  376 CONTINUE                                                          SETC3480
C*** TEST FOR ERROR FREE SETUP.  PUT POINTER IN SYSTEM LIST.             SETC3490
  378 IF (NFLAG3.LE.0)   KRST(KT) = KRST(KT) + NMSL                     SETC3500
  380 CONTINUE                                                          SETC3510
  381 IF (NMSL .LE. 0) GO TO 400                                        SETC3520
C    UNPACK DATMSL                                                      SETC3530
      DO 382 I = 1, NMSL                                                SETC3540
      K = DATMSL(2,I) * 100. + .4                                       SETC3550
      DATMSL(2,I) = K / 100                                             SETC3560
      DATMSL(13,I) = MOD(K,100)                                         SETC3570
      K = DATMSL(3,I) * 10. + .4                                        SETC3580
      DATMSL(3,I) = K / 10                                              SETC3590
      DATMSL(14,I) = MOD(K,10)                                          SETC3600
      K = DATMSL(7,I) * 100. + .4                                       SETC3610
      DATMSL(7,I) = K / 100                                             SETC3620
      DATMSL(15,I) = MOD(K,100)                                         SETC3630
      DATMSL(15,I) = DATMSL(15,I) / 100.                                SETC3640
  382 DATMSL(6,I) = DATMSL(6,I) / 60.                                   SETC3650
C***  ADJUST FOR SALVO SIZE                                             SETC3660
      DO 386 KT = 1, NBSS                                               SETC3670
      K = KBST(KT) / 1000000                                            SETC3680
      IF (K .NE. 76 .AND. K .NE. 78) GO TO 386                          SETC3690
      K = MOD(KBST(KT),100)                                             SETC3700
      DO 384 KE = 1, NBP                                                SETC3710
      IF (KBPXS(KT,KE) .EQ. 0) GO TO 384                                SETC3720
      LW1 = (KBPXS(KT,KE) / 10000) * 10000                              SETC3730
      MW1 = KBPXS(KT,KE) - LW1                                          SETC3740
      LW2 = MW1 / 100                                                   SETC3750
      MW2 = MW1 - LW2 * 100                                             SETC3760
      LW3 = LW2 * MW2                                                   SETC3770
      IF (LW2 .EQ. 99 .OR. LW3 .EQ. 0) GO TO 384                        SETC3780
      MW3 = DATMSL(5,K) + .4                                            SETC3790
      LW3 = LW3 / MW3                                                   SETC3800
      KBPXS(KT,KE) = LW1 + LW3 * 100 + MW2                              SETC3810
  384 CONTINUE                                                          SETC3820
  386 CONTINUE                                                          SETC3830
      DO 390 KT = 1, NRSS                                               SETC3840
      K = KRST(KT) / 1000000                                            SETC3850
      IF (K .NE. 96 .AND. K .NE. 98) GO TO 390                          SETC3860
      K = MOD(KRST(KT),100)                                             SETC3870
      DO 388 KE = 1, NRP                                                SETC3880
      IF (KRPXS(KT,KE) .EQ. 0) GO TO 388                                SETC3890
      LW1 = (KRPXS(KT,KE) / 10000) * 10000                              SETC3900
      MW1 = KRPXS(KT,KE) - LW1                                          SETC3910
      LW2 = MW1 / 100                                                   SETC3920
      MW2 = MW1 - LW2 * 100                                             SETC3930
      LW3 = LW2 * MW2                                                   SETC3940
      IF (LW2 .EQ. 99 .OR. LW3 .EQ. 0) GO TO 388                        SETC3950
      MW3 = DATMSL(5,K) + .4                                            SETC3960
      LW3 = LW3 / MW3                                                   SETC3970
      KRPXS(KT,KE) = LW1 + LW3 * 100 + MW2                              SETC3980
  388 CONTINUE                                                          SETC3990
  390 CONTINUE                                                          SETC4000
C*** WRAP UP                                                            SETC4010
  400 IF (NFLAG2.GT.0)  NFLAG=NFLAG + 1                                 SETC4020
      IF (IPRINT.GT.0)  GO TO 430                                       SETC4030
  402 WRITE(N6,2011) NFLAG                                              SETC4040
      RETURN                                                            SETC4050
C***  PRINT OPTIONS             ***                 ***                 SETC4060
C                                                                       SETC4070
      ENTRY         PRSETC(IPRINT)                                      SETC4080
C                                                                       SETC4090
C***  SSMS & ASMS                                                       SETC4100
  430 CALL  PAGE                                                        SETC4110
      WRITE(N6,3020)  NBSSM, NBASM, NRSSM, NRASM                        SETC4120
      IF (NMSL.LE.0)  GO TO 436                                         SETC4130
      IF (IPRINT.LT.2)  GO TO 436                                       SETC4140
      DO 432 KT=1,NMSL                                                  SETC4150
      WRITE(N6,3022) KT, (NAMMSL(I,KT),I=1,2),KMSLTY(KT)                SETC4160
  432 WRITE(N6,3024) (DATMSL(I,KT),I=1,LATMSL)                          SETC4170
      CALL  PAGE                                                        SETC4180
      DO 434 KT=1,NMSL                                                  SETC4190
      WRITE(N6,3023) KT, (NAMMSL(I,KT),I=1,2),KMSLTY(KT)                SETC4200
  434 WRITE(N6,3024) (TTRAJ(I,KT),I=1,LTRAJ)                            SETC4210
  436 WRITE(N6,2011) NFLAG                                              SETC4220
      RETURN                                                            SETC4230
      END                                                               SETC4240


      SUBROUTINE STTRAJ(KT,KTEST,KL,VW1,LA,MA,IXX,TTRAJ,LTRAJ)          STTR0010
C  PGM=NXX(NEM). L.D.G.                    9-15-73    FORTRAN IV    EBCD  STTR0020
C  TO SETUP TEMPORARY TRAJECTORIES FOR MSLS & A/C.                      STTR0030
C                                                                       STTR0040
C***  KT     =  1,   PRIMARY TRAJ                                       STTR0050
C            =  2    SHORT RANGE TRAJ                                   STTR0060
C     KTEST =  TRAJ CODE IN TABLE 4110 OR 5110, E.G. 401=PRIMARY TRAJ=4 STTR0070
C     KL     =  LINE NO.IN VW1 FOR TRAJ TO BE SET UP.                   STTR0080
C     VW1(KL,J) = JTH ITEM IN LINE=KL                                   STTR0090
C     LA, MA=  ABSOLUTE DIMENSIONS OF VW1                               STTR0100
C     IXX    =  ERROR INDEX.  =0 IF OK   = 1 IF KO.                     STTR0110
C     TTRAJ =  ARRAY OF TRAJ ELEMENTS                                   STTR0120
C     LTRAJ =  LENGTH OF TTRAJ                                          STTR0130
C***                                                                    STTR0140
      DIMENSION  VW1(LA,MA), TTRAJ(LTRAJ)                               STTR0150
 1    IXX = 0                                                           STTR0160
      J   = 0                                                           STTR0170
      IF (KT.EQ.2)  J=25                                                STTR0180
      TTRAJ(J+1) = KTEST                                                STTR0190
      TTRAJ(J+2) = VW1(KL,1)                                            STTR0200
      KD  = VW1(KL,2)*1000. + .4                                        STTR0210
      TTRAJ(J+3) = KD / 1000                                            STTR0220
      TTRAJ(J+4) =  MOD(KD,1000)                                        STTR0230
      K    = 1                                                          STTR0240
      RCUM= .0                                                          STTR0250
      DO 18 L=1,5                                                       STTR0260
      K    = K + 2                                                      STTR0270
      M    = J + 4*L                                                    STTR0280
      H1   = VW1(KL,K+1) / 6.080                                        STTR0290
      H2   = .0                                                         STTR0300
      IF (L.LE.4)   H2=VW1(KL,K+3) / 6.080                              STTR0310
      KD   = ABS(VW1(KL,K)*10000.) + .4                                 STTR0320
      R1   = KD / 10000                                                 STTR0330
      V1   = MOD(KD,10000)                                              STTR0340
      D    = SQRT((H2-H1)**2 + R1**2)                                   STTR0350
      IF (D.LT..001)   GO TO 10                                         STTR0360
      SINA = (H2-H1) / D                                                STTR0370
      COSA = R1 / D                                                     STTR0380
      VH1  = V1*COSA                                                    STTR0390
      VV1  = V1*SINA                                                    STTR0400
      GO TO 12                                                          STTR0410
C*** ZERO LENGTH LEG                                                     STTR0420
 10   VH1 = .001                                                        STTR0430
      VV1 = .0                                                          STTR0440
      R1  = .0001                                                       STTR0450
 12   RCUM= RCUM + R1                                                   STTR0460
C*** TAKE CARE OF AIRPLANE FLAG                                          STTR0470
      IF (VW1(KL,K).GT.(-.0001))  GO TO 14                              STTR0480
      VH1 = -V1                                                         STTR0490
      VV1 = .0                                                          STTR0500
      H2  = H1                                                          STTR0510
 14   TTRAJ(M+1) = H1                                                   STTR0520
      TTRAJ(M+2) = RCUM                                                 STTR0530
      TTRAJ(M+3) = VH1                                                  STTR0540
      TTRAJ(M+4) = VV1                                                  STTR0550
 18   CONTINUE                                                          STTR0560
C*** FILL EXTRA POINT                                                    STTR0570
      TTRAJ(J+25) = .0                                                  STTR0580
C                                                                       STTR0590
C*** REVERSE CUM RANGE                                                   STTR0600
      DO 22 L=2,5                                                       STTR0610
      M  = J - 4*L + 28                                                 STTR0620
 22   TTRAJ(M+2) = RCUM - TTRAJ(M-2)                                    STTR0630
      TTRAJ(J+6) = RCUM                                                 STTR0640
      IF (TTRAJ(J+2).LT.100..OR.TTRAJ(J+3).LT..001)  IXX=1              STTR0650
      RETURN                                                            STTR0660
      END                                                               STTR0670


      SUBROUTINE  SETUPD(IPRINT)                                        SETD0010
C  PGM=NXX(NEM). E.C.     VER.1   10-18-73          FORTRAN IV    EBCD  SETD0020
C  MODS.RED AAM 9900=420-451.                         VER.2   12-19-74 SETD0030
C  TO SET UP SYSTEMS DATA FOR CURRENT PROBLEM FOR SYSTEMS IN SETUPB     SETD0040
C  CALLED BY MAIN AND ZIP CARD AS FOLLOWS                               SETD0050
CIP  8 4 P                              SETUPD. MISC. WEAPONS           SETD0060
C***                                                                    SETD0070
     COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              SETD0080
```

```
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                      SETD0090
     1,IDUMB( 72), NFLAG, NFLAG2                                        SETD0100
CZAZA                                                                   SETD0110
      INTEGER TITLA                                                     SETD0120
      COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)     SETD0130
     1, TITLA(15),HEADA(37),IFMT                                        SETD0140
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                          SETD0150
      DIMENSION  BZA( 555)                                              SETD0160
      EQUIVALENCE  (NCBA,BZA(1))                                        SETD0170
CNAVIG                                                                  SETD0180
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                  SETD0190
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),      SETD0200
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),      SETD0210
     3          TTIME,NUMX,                                             SETD0220
     4     KBU,KBUK(4,40),BREL(4,40),BXYZ(40,7),NAMBU(40,2),BV(40,8),   SETD0230
     5     KRU,KRUK(4,40),RREL(4,40),RXYZ(40,7),NAMRU(40,2),RV(40,8)    SETD0240
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                     SETD0250
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),           SETD0260
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),         SETD0270
     2  BPX1(12,15),                                                    SETD0280
     3                NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),        SETD0290
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),         SETD0300
     5  RPX1(12,15)                                                     SETD0310
      COMMON/CWORK/ LA2,MA2,LA3,LA4,MA4,KA2(43),A,VA2(179),K,           SETD0320
     1 KA5(1170)                                                        SETD0330
CZZSYS                              25      11      12    15            SETD0340
      COMMON/CZZSYS/NZZSYS,NZZSMX,NBZSYS,NRZSYS,LAZZ,NAMSYS(2,25),      SETD0350
     1    KZZTYP(25),ZZSYS(15,25)                                       SETD0360
CSAMLT              8,12,15,                                            SETD0370
      COMMON/CSAMLT/L,M,N,RMN(15),RMX(15),EMN(15),EMX(15),ZMX(15),      SETD0380
     1 XMX(15),  RT(8,12,15)                                            SETD0390
CSASYS                               15      4      4      2     4    20  SETD0400
      COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,       SETD0410
     1    NAMSA(2,15),KSATYP(15),SASYS(20,15)                           SETD0420
      DIMENSION KBZZT(5), KRZZT(5), KBSAST(2), KRSAST(2)                SETD0430
      DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/                            SETD0440
      DATA NBZZT, NRZZT, KBZZT, KRZZT/5, 5, 71000000, 72000000,         SETD0450
     1 73000000,74000000,79000000,91000000,92000000,93000000,94000000,  SETD0460
     2 99000000/                                                        SETD0470
      DATA NBSAST, NRSAST, KBSAST, KRSAST/2, 2, 75100000, 77100000,     SETD0480
     1 95100000,97100000/                                               SETD0490
      DATA NLTMX, NMTMX, NNTMX/8, 12, 15/                               SETD0500
 5000 FORMAT(1H0,'NO TABLE FOUND FOR ID1, ID2=', 2I10)                  SETD0510
 5010 FORMAT(1H0,'NO. OF MISC. WEAPON SYSTEMS=',I4,', EXCEEDS STORAGE='SETD0520
     1,I4)                                                              SETD0530
 5020 FORMAT(1H0,'NO. OF SURFACE-TO-AIR SYSTEMS=',I4,', EXCEEDS STORAGESETD0540
     1=',I4)                                                            SETD0550
 5030 FORMAT(1H0,'NO. TIME LINE TABLES=',I4,', EXCEEDS STORAGE=',I4)    SETD0560
 5040 FORMAT(6X,'AAGUNS & SAMS'/                                        SETD0570
     1    8X,'NO.BLU GUN TYPES=',I4,',  NO.BLU SAM TYPES=',I4/          SETD0580
     2    8X,'NO.RED GUN TYPES=',I4,',  NO.RED SAM TYPES=',I4)          SETD0590
 5050 FORMAT(/8X,'SYSTEM',13X,'MIN RANGE   MAX RANGE   MIN ELEV    MAX SETD0600
     1ELEV    MAX ALT     MAX HORIZ.'/ 8X,'NAME AND TYPE', 8X,'NM',     SETD0610
     2 10X,'NM', 10X,'DEG', 9X,'DEG', 9X,'KFT', 9X,'NM')                SETD0620
 5060 FORMAT(1X, I6,'.', 2A4, I9, 8G12.5)                               SETD0630
 5070  FORMAT(1H0,'AT END OF SETUPD,CUM NO. SETUP ERRORS=NFLAG=',I5)       SETD0640
 5080  FORMAT(6X,'MISC. WEAPON DATA')                                     SETD0650
 5090  FORMAT(/I7,'.',2A4,I9,'     WEAPON DATA')                          SETD0660
 5100  FORMAT(8X,10G12.5)                                                 SETD0670
 5110  FORMAT(6X,'AAGUNS & SAMS')                                         SETD0680
 5120  FORMAT(6X,'MISC. WEAPON DATA'/                                     SETD0690
     1     8X,'SYSTEM',13X,'MAX RANGE    VELOCITY     WHD WT'/            SETD0700
     2     8X,'NAME AND TYPE',8X,'NM', 10X,'KTS', 9X'LBS')                SETD0710
 5130  FORMAT(I7,'.',2A4,I9,8G12.5)                                       SETD0720
       KZARL = 96 + LA*(5 + MA)                                           SETD0730
       JZARL = KZARL                                                      SETD0740
       NFLAG2 = 0                                                         SETD0750
       NZZSMX = 25                                                        SETD0760
       LAZZ = 15                                                          SETD0770
       NSASMX = 15                                                        SETD0780
       LASA = 20                                                          SETD0790
       NZZSYS = 0                                                         SETD0800
       NBZSYS = 0                                                         SETD0810
       NRZSYS = 0                                                         SETD0820
       L = NLTMX                                                          SETD0830
       M = NMTMX                                                          SETD0840
       N = NNTMX                                                          SETD0850
       NSASYS = 0                                                         SETD0860
       NBGUN  = 0                                                         SETD0870
       NBSAM  = 0                                                         SETD0880
       NRGUN  = 0                                                         SETD0890
       NRSAM  = 0                                                         SETD0900
       IY = 3                                                             SETD0910
       ID2 = -1                                                           SETD0920
C***  SETUP BLU MISCELLANEOUS WEAPON SYSTEMS                               SETD0930
       IF (KBU .LE. 0) GO TO 400                                          SETD0940
C***  CYCLE THRU BLU SYSTEMS                                               SETD0950
       DO  70 KC = 1, NBZZT                                               SETD0960
       ID1 = KBZZT(KC)                                                    SETD0970
       CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                   SETD0980
       IF (IXX .EQ. 0) GO TO  10                                          SETD0990
       WRITE(N6,5000) ID1,ID2                                             SETD1000
       GO TO  70                                                          SETD1010
   10  DO  60 KT = 1, NBSS                                                SETD1020
       K = KBST(KT) / 10000                                               SETD1030
       IF (ID1/1000000 .NE. K/100) GO TO  60                              SETD1040
       DO  20 J = 1, LA1                                                  SETD1050
       IF (K .EQ. KA1(J,5)) GO TO  30                                     SETD1060
   20  CONTINUE                                                           SETD1070
       GO TO  60                                                          SETD1080
C***  FOUND SYSTEM IN CATALOG                                              SETD1090
   30  JT = J                                                             SETD1100
       NBZSYS = NBZSYS + 1                                                SETD1110
       IF (NBZSYS .LE. NZZSMX) GO TO  40                                  SETD1120
       WRITE(N6,5010) NBZSYS, NZZSMX                                      SETD1130
       NFLAG2 = NFLAG2 + 1                                                SETD1140
       NBZSYS = NZZSMX                                                    SETD1150
       GO TO 100                                                          SETD1160
C***  TRANSFER SYSTEM NAME AND DATA.  PUT POINTER IN KBST.                 SETD1170
   40  NABSS(1,KT) = KA1(JT,2)                                            SETD1180
```

```
      NAMSYS(1,NBZSYS) = KA1(JT,2)                                      SETD1190
      NABSS(2,KT) = KA1(JT,3)                                           SETD1200
      NAMSYS(2,NBZSYS) = KA1(JT,3)                                      SETD1210
      KZZTYP(NBZSYS) = K * 10000                                        SETD1220
      KBST(KT) = K * 10000 + NBZSYS                                     SETD1230
      DO  50 KD = 1, 12                                                 SETD1240
   50 ZZSYS(KD,NBZSYS) = VA1(JT,KD)                                     SETD1250
      A = IFIX(VA1(JT,2) + .004)                                        SETD1260
      ZZSYS( 2,NBZSYS) = A                                              SETD1270
      ZZSYS(13,NBZSYS) = IFIX((VA1(JT,2)-A) * 100. + .4)                SETD1280
      A = IFIX(VA1(JT,3) + .004)                                        SETD1290
      ZZSYS( 3,NBZSYS) = A                                              SETD1300
      ZZSYS(14,NBZSYS) = IFIX((VA1(JT,3)-A) * 10. + .4)                 SETD1310
      A = IFIX(VA1(JT,7) + .004)                                        SETD1320
      ZZSYS( 7,NBZSYS) = A                                              SETD1330
      ZZSYS(15,NBZSYS) = VA1(JT,7) - A                                  SETD1340
      ZZSYS(6,NBZSYS) = ZZSYS(6,NBZSYS) / 60.                           SETD1350
C***  ADJUST FOR SALVO SIZE                                             SETD1360
      MA4 = VA1(JT,5) + .00001                                          SETD1370
      DO  55 KE = 1, NBP                                                SETD1380
      IF (KBPXS(KT,KE) .EQ. 0) GO TO  55                                SETD1390
      LA2 = (KBPXS(KT,KE) / 10000) * 10000                              SETD1400
      MA2 = KBPXS(KT,KE) - LA2                                          SETD1410
      LA3 = MA2 / 100                                                   SETD1420
      MA3 = MA2 - LA3 * 100                                             SETD1430
      LA4 = LA3 * MA3                                                   SETD1440
      IF (LA3 .EQ. 99 .OR. LA4 .EQ. 0) GO TO  55                        SETD1450
      LA4 = LA4 / MA4                                                   SETD1460
      KBPXS(KT,KE) = LA2 + LA4*100 + MA3                                SETD1470
   55 CONTINUE                                                          SETD1480
   60 CONTINUE                                                          SETD1490
   70 CONTINUE                                                          SETD1500
  100 NZZSYS = NBZSYS                                                   SETD1510
C***  SETUP BLU SURFACE-TO-AIR SYSTEMS                                  SETD1520
      DO 170 KC = 1, NBSAST                                             SETD1530
      ID1 = KBSAST(KC)                                                  SETD1540
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                  SETD1550
      IF (IXX .EQ. 0) GO TO 110                                         SETD1560
      WRITE(N6,5000) ID1,ID2                                            SETD1570
      GO TO 170                                                         SETD1580
  110 DO 160 KT = 1, NBSS                                               SETD1590
      K = KBST(KT) / 10000                                              SETD1600
      IF (ID1/1000000 .NE. K/100) GO TO 160                             SETD1610
      DO 120 J = 1, LA1                                                 SETD1620
      IF (K .EQ. KA1(J,5)) GO TO 130                                    SETD1630
  120 CONTINUE                                                          SETD1640
      GO TO 160                                                         SETD1650
C***  FOUND SYSTEM IN CATALOG                                           SETD1660
  130 JT = J                                                            SETD1670
      GO TO (131, 132), KC                                              SETD1680
  131 NBGUN = NBGUN + 1                                                 SETD1690
      GO TO 133                                                         SETD1700
  132 NBSAM = NBSAM + 1                                                 SETD1710
  133 NSASYS = NSASYS + 1                                               SETD1720
      IF (NSASYS .LE. NSASMX) GO TO 140                                 SETD1730
      WRITE(N6,5020) NSASYS, NSASMX
      NFLAG2 = NFLAG2 + 1
      NSASYS = NSASMX
      GO TO (134, 135), KC
  134 NBGUN = NBGUN - 1
      GO TO 400
  135 NBSAM = NBSAM - 1
      GO TO 400
C***  TRANSFER SYSTEM NAME AND DATA.  PUT POINTER IN KBST.
  140 NABSS(1,KT) = KA1(JT,2)
      NAMSA(1,NSASYS) = KA1(JT,2)
      NABSS(2,KT) = KA1(JT,3)
      NAMSA(2,NSASYS) = KA1(JT,3)
      KSATYP(NSASYS) = K * 10000
      KBST(KT) = K * 10000 + NSASYS
      DO 150 KD = 1, 12
  150 SASYS(KD,NSASYS) = VA1(JT,KD)
      A = IFIX(VA1(JT,2) + .004)
      SASYS( 2,NSASYS) = A
      SASYS(13,NSASYS) = IFIX((VA1(JT,2)-A) * 100. + .4)
      A = IFIX(VA1(JT,3) + .004)
      SASYS( 3,NSASYS) = A
      SASYS(14,NSASYS) = IFIX((VA1(JT,3)-A) * 10. + .4)
      A = IFIX(VA1(JT,7) + .004)
      SASYS( 7,NSASYS) = A
      SASYS(15,NSASYS) = VA1(JT,7) - A
      SASYS(6,NSASYS) = SASYS(6,NSASYS) / 60.
C***  ADJUST FOR SALVO SIZE
      MA4 = VA1(JT,5) + .00001
      DO 155 KE = 1, NBP
      IF (KBPXS(KT,KE) .EQ. 0) GO TO 155
      LA2 = (KBPXS(KT,KE) / 10000) * 10000
      MA2 = KBPXS(KT,KE) - LA2
      LA3 = MA2 / 100
      MA3 = MA2 - LA3 * 100
      LA4 = LA3 * MA3
      IF (LA3 .EQ. 99 .OR. LA4 .EQ. 0) GO TO 155
      LA4 = LA4 / MA4
      KBPXS(KT,KE) = LA2 + LA4*100 + MA3
  155 CONTINUE
  160 CONTINUE
  170 CONTINUE
C***  SETUP RED MISCELLANEOUS WEAPON SYSTEMS
  400 IF (KRU .LE. 0) GO TO 600
C***  CYCLE THRU RED SYSTEMS
      DO 470 KC = 1, NRZZT
      ID1 = KRZZT(KC)
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX .EQ. 0) GO TO 410
      WRITE(N6,5000) ID1,ID2
      GO TO 470
  410 DO 460 KT = 1, NRSS
      K = KRST(KT) / 10000
      IF (ID1/1000000 .NE. K/100) GO TO 460
      DO 420 J = 1, LA1
      IF (K  .EQ. KA1(J,5)) GO TO 430                                   SETD2290
  420 CONTINUE                                                          SETD2300
      GO TO 460                                                         SETD2310
C***  FOUND SYSTEM IN CATALOG                                           SETD2320
  430 JT = J                                                            SETD2330
      NZZSYS = NZZSYS + 1                                               SETD2340
      IF (NZZSYS .LE. NZZSMX) GO TO 440                                 SETD2350
      WRITE(N6,5010) NZZSYS, NZZSMX                                     SETD2360
      NFLAG2 = NFLAG2 + 1                                               SETD2370
      NZZSYS = NZZSMX                                                   SETD2380
      GO TO 500                                                         SETD2390
C***  TRANSFER SYSTEM NAME AND DATA.  PUT POINTER IN KRST.              SETD2400
  440 NARSS(1,KT) = KA1(JT,2)                                           SETD2410
      NAMSYS(1,NZZSYS) = KA1(JT,2)                                      SETD2420
      NARSS(2,KT) = KA1(JT,3)                                           SETD2430
      NAMSYS(2,NZZSYS) = KA1(JT,3)                                      SETD2440
      KZZTYP(NZZSYS) = K * 10000                                        SETD2450
      KRST(KT) = K * 10000 + NZZSYS                                     SETD2460
      DO 450 KD = 1, 12                                                 SETD2470
  450 ZZSYS(KD,NZZSYS) = VA1(JT,KD)                                     SETD2480
      A = IFIX(VA1(JT,2) + .004)                                        SETD2490
      ZZSYS( 2,NZZSYS) = A                                              SETD2500
      ZZSYS(13,NZZSYS) = IFIX((VA1(JT,2)-A) * 100. + .4)                SETD2510
      A = IFIX(VA1(JT,3) + .004)                                        SETD2520
      ZZSYS( 3,NZZSYS) = A                                              SETD2530
      ZZSYS(14,NZZSYS) = IFIX((VA1(JT,3)-A) * 10. + .4)                 SETD2540
      A = IFIX(VA1(JT,7) + .004)                                        SETD2550
      ZZSYS( 7,NZZSYS) = A                                              SETD2560
      ZZSYS(15,NZZSYS) = VA1(JT,7) - A                                  SETD2570
      ZZSYS(6,NZZSYS) = ZZSYS(6,NZZSYS) / 60.                           SETD2580
C***  ADJUST FOR SALVO SIZE                                             SETD2590
      MA4 = VA1(JT,5) + .00001                                          SETD2600
      DO 455 KE = 1, NRP                                                SETD2610
      IF (KRPXS(KT,KE) .EQ. 0) GO TO 455                                SETD2620
      LA2 = (KRPXS(KT,KE) / 10000) * 10000                              SETD2630
      MA2 = KRPXS(KT,KE) - LA2                                          SETD2640
      LA3 = MA2 / 100                                                   SETD2650
      MA3 = MA2 - LA3 * 100                                             SETD2660
      LA4 = LA3 * MA3                                                   SETD2670
      IF (LA3 .EQ. 99 .OR. LA4 .EQ. 0) GO TO 455                        SETD2680
      LA4 = LA4 / MA4                                                   SETD2690
      KRPXS(KT,KE) = LA2 + LA4*100 + MA3                                SETD2700
  455 CONTINUE                                                          SETD2710
  460 CONTINUE                                                          SETD2720
  470 CONTINUE                                                          SETD2730
  500 NRZSYS = NZZSYS - NBZSYS                                          SETD2740
C***  SETUP RED SURFACE-TO-AIR SYSTEMS                                  SETD2750
      DO 570 KC = 1, NRSAST                                             SETD2760
      ID1 = KRSAST(KC)                                                  SETD2770
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                  SETD2780
      IF (IXX .EQ. 0) GO TO 510                                         SETD2790
      WRITE(N6,5000) ID1,ID2                                            SETD2800
      GO TO 570                                                         SETD2810
  510 DO 560 KT = 1, NRSS                                               SETD2820
      K = KRST(KT) / 10000                                              SETD2830
      IF (ID1/100000C .NE. K/100) GO TO 560
      DO 520 J = 1, LA1
      IF (K .EQ. KA1(J,5)) GO TO 530
  520 CONTINUE
      GO TO 560
C***  FOUND SYSTEM IN CATALOG
  530 JT = J
      GO TO (531, 532), KC
  531 NRGUN = NRGUN + 1
      GO TO 533
  532 NRSAM = NRSAM + 1
  533 NSASYS = NSASYS + 1
      IF (NSASYS .LE. NSASMX) GO TO 540
      WRITE(N6,5020) NSASYS, NSASMX
      NFLAG2 = NFLAG2 + 1
      NSASYS = NSASMX
      GO TO (534, 535), KC
  534 NRGUN = NRGUN - 1
      GO TO 600
  535 NRSAM = NRSAM - 1
      GO TO 600
C***  TRANSFER SYSTEM NAME AND DATA.  PUT POINTER IN KRST.
  540 NARSS(1,KT) = KA1(JT,2)
      NAMSA(1,NSASYS) = KA1(JT,2)
      NARSS(2,KT) = KA1(JT,3)
      NAMSA(2,NSASYS) = KA1(JT,3)
      KSATYP(NSASYS) = K * 10000
      KRST(KT) = K * 10000 + NSASYS
      DO 550 KD = 1, 12
  550 SASYS(KD,NSASYS) = VA1(JT,KD)
      A = IFIX(VA1(JT,2) + .004)
      SASYS( 2,NSASYS) = A
      SASYS(13,NSASYS) = IFIX((VA1(JT,2)-A) * 100. + .4)
      A = IFIX(VA1(JT,3) + .004)
      SASYS( 3,NSASYS) = A
      SASYS(14,NSASYS) = IFIX((VA1(JT,3)-A) * 10. + .4)
      A = IFIX(VA1(JT,7) + .004)
      SASYS( 7,NSASYS) = A
      SASYS(15,NSASYS) = VA1(JT,7) - A
      SASYS(6,NSASYS) = SASYS(6,NSASYS) / 60.
C***  ADJUST FOR SALVO SIZE
      MA4 = VA1(JT,5) + .00001
      DO 555 KE = 1, NRP
      IF (KRPXS(KT,KE) .EQ. 0) GO TO 555
      LA2 = (KRPXS(KT,KE) / 10000) * 10000
      MA2 = KRPXS(KT,KE) - LA2
      LA3 = MA2 / 100
      MA3 = MA2 - LA3 * 100
      LA4 = LA3 * MA3
      IF (LA3 .EQ. 99 .OR. LA4 .EQ. 0) GO TO 555
      LA4 = LA4 / MA4
      KRPXS(KT,KE) = LA2 + LA4*100 + MA3
  555 CONTINUE
  560 CONTINUE
  570 CONTINUE
```

```
C***  SETUP TIME LINES 75XX, 77XX, 95XX, 97XX                           SETD3390
  600 IF (NSASYS .LE. 0) GO TO 700                                      SETD3400
      DO 670 KC = 1, NSASYS                                             SETD3410
      ID1 = KSATYP(KC)                                                  SETD3420
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                  SETD3430
      IF (IXX .EQ. 0) GO TO 610                                         SETD3440
      WRITE(N6,5000) ID1,ID2                                            SETD3450
      GO TO 670                                                         SETD3460
  610 DO 620 I = 1, LA1                                                 SETD3470
      DO 620 J = 1, MA1                                                 SETD3480
  620 RT(I,J,KC) = VA1(I,J)                                             SETD3490
      RMN(KC) = CA(4)                                                   SETD3500
      RMX(KC) = CA(8)                                                   SETD3510
      EMN(KC) = ICA(4)                                                  SETD3520
      EMX(KC) = ICA(8)                                                  SETD3530
      ZMX(KC) = ICA(16)                                                 SETD3540
      XMX(KC) = CA(16)                                                  SETD3550
      SASYS(16,KC) = RMN(KC)                                            SETD3560
      SASYS( 1,KC) = RMX(KC)                                            SETD3570
  670 CONTINUE                                                          SETD3580
  700 IF (NFLAG2 .GT. 0) NFLAG = NFLAG + 1                              SETD3590
      IF (IPRINT .GT. 0) GO TO 800                                      SETD3600
      GO TO 900                                                         SETD3610
C***  PRINT OPTIONS                          * * *                      SETD3620
C***                                         * * *                      SETD3630
      ENTRY       PRSETD(IPRINT)                                        SETD3640
C***                                         * * *                      SETD3650
  800 IF (NZZSYS .LE. 0) GO TO 808                                      SETD3660
      CALL PAGE                                                         SETD3670
      WRITE(N6,5120)                                                    SETD3680
      DO 804 I = 1, NZZSYS                                              SETD3690
  804 WRITE(N6,5130) I, (NAMSYS(J,I),J=1,2), KZZTYP(I),                 SETD3700
     1     (ZZSYS(J,I),J=1,3)                                           SETD3710
      IF (IPRINT .LT. 2) GO TO 808                                      SETD3720
      CALL PAGE                                                         SETD3730
      WRITE(N6,5080)                                                    SETD3740
      DO 806 I = 1, NZZSYS                                              SETD3750
      WRITE(N6,5090) I, (NAMSYS(J,I),J=1,2), KZZTYP(I)                  SETD3760
  806 WRITE(N6,5100) (ZZSYS(J,I),J=1,LAZZ)                              SETD3770
  808 IF (NSASYS .LE. 0) GO TO  900                                     SETD3780
      CALL PAGE                                                         SETD3790
      WRITE(N6,5040) NBGUN, NBSAM, NRGUN, NRSAM                         SETD3800
      WRITE(N6,5050)                                                    SETD3810
      DO 810 I = 1, NSASYS                                              SETD3820
  810 WRITE(N6,5060) I, (NAMSA(J,I),J=1,2), KSATYP(I), RMN(I), RMX(I),  SETD3830
     1 EMN(I), EMX(I), ZMX(I), XMX(I)                                   SETD3840
      IF (IPRINT .LT. 2) GO TO 900                                      SETD3850
      CALL PAGE                                                         SETD3860
C***  PRINT TIME LINES                                                  SETD3870
      DO 820 I = 1, NSASYS                                              SETD3880
      LA2 = RT(1,1,I) + .1                                              SETD3890
      MA2 = MOD(LA2,100)                                                SETD3900
      LA2 = LA2 / 100                                                   SETD3910
  820 CALL WMAT3(2,K,RT,L,M,N,LA2,MA2,I,NLINE,60HBORDERED TABLE, AAGUN  SETD3920
     1OR SAM TIME OF FLIGHT (SEC VS DEG)      )                         SETD3930
  850   CALL PAGE                                                            SETD3940
        WRITE(N6,5110)                                                       SETD3950
        DO 860 I = 1, NSASYS                                                 SETD3960
        WRITE(N6,5090) I, (NAMSA(J,I),J=1,2), KSATYP(I)                     SETD3970
  860   WRITE(N6,5100) (SASYS(J,I),J=1,LASA)                                 SETD3980
  900 WRITE(N6,5070) NFLAG                                                   SETD3990
        RETURN                                                               SETD4000
        END                                                                  SETD4010


        SUBROUTINE SETUPE(IPRINT)                                            SETE0010
C   PGM=NXX(NEM). L.D.G.         VER.5.  9-7-73         FORTRAN IV     EBCD  SETE0020
C   TO SET UP SYSTEMS DATA FOR CURRENT PROBLEM FOR SYSTEMS IN SETUPB         SETE0030
C   CALLED BY MAIN AND ZIP CARD AS FOLLOWS                                   SETE0040
CIP   8 5 P                                   SETUPE. RADARS AND JAMMERS     SETE0050
C***                                                                         SETE0060
       COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12                 SETE0070
        COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                         SETE0080
     1,IDUMB(72), NFLAG, NFLAG2                                              SETE0090
CZAZA                                                                        SETE0100
        INTEGER TITLA                                                        SETE0110
        COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)        SETE0120
     1, TITLA(15),HEADA(37),IFMT                                             SETE0130
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                               SETE0140
        DIMENSION  BZA( 555)                                                 SETE0150
        EQUIVALENCE  (NCBA,BZA(1))                                           SETE0160
        DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/                               SETE0170
CNAVIG                                                                       SETE0180
        COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                     SETE0190
     1      NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),          SETE0200
     2      NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),          SETE0210
     3           TTIME,NUMX,                                                 SETE0220
     4      KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),       SETE0230
     5      KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)        SETE0240
        COMMON/CWORK/LA2,MA2,LAA2,MAA2, KA2(3,15),VA2(12,15),                SETE0250
     1               LA3,MA3,LAA3,MAA3, VA3(36,15),              KA5( 727)   SETE0260
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                          SETE0270
        COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),              SETE0280
     1    NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),            SETE0290
     2    BPX1(12,15),                                                       SETE0300
     3                 NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),            SETE0310
     4    NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),            SETE0320
     5    RPX1(12,15)                                                        SETE0330
CRADAR                                                                       SETE0340
        COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                             SETE0350
     1    BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,3,2),             SETE0360
     2                NRSR,NRTR,NRRMX,NRJ,NRJMX,                             SETE0370
     3    RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,3,2)              SETE0380
C***                                                                         SETE0390
C   NBSR    = NO.BLU SEARCH RADARS          / NRSR    = NO.RED SEARCH RADARS SETE0400
C   NBTR    = NO.BLU TRACK RADARS           / NRTR    = NO.RED TRACK RADARS  SETE0410
C   NBRMX   = MAX NO.BLU RADARS             / NRRMX   = MAX NO.RED RADARS    SETE0420
C   BRAD(I,J)= BLU, ITH DATA ON JTH RADAR/ RRAD(I,J)=  ETC.                  SETE0430
C   BENV(I,K)= BLU RADAR ENVIRONMENT K    / RENV(I,K)=                       SETE0440
```

```
C   BETC(I,L)= BLU MISC RADAR DATA   L      /    ETC.
C   BTAR(I,M)= BLU RADAR TARGET DATA  M     /
C   BJAM(I,J,N)=BLU JAMMER DATA N           /
C   NOTE. THE POINTER TO THE JTH RADAR IS/
C      IN KBST( )=KBST( )+J,(J.LE.99).      /
C     & NTH JAMMER IN KBST( )=KBST( )+N     /
C***
C
       DIMENSION  KBRTT( 2), KRRTT( 2)
      DIMENSION PCOL(2),PRTPE(2,2)
       DATA  NBRTT,KBRTT/ 2,65100000,67100000/
       DATA  NRRTT,KRRTT/ 2,85100000,87100000/
      DATA PCOL/' BLU',' RED'/
      DATA PRTPE/' SEA','RCH ',' TRA','CK  '/
CSONAR                          15     5     8    12
       COMMON/CSONAR/NSON,NSONMX,NBSON,NRSON,LASON,NAMSON( 2,15),
     1    KSONTY(15), SONSYS(12,15)
 2005  FORMAT(1H0,22HERROR IN SETUPE AT E1=,F8.2,3X,'ID1=',2I10,2G13.6)
 2006  FORMAT(1H0,'NO. OF RADARS=',I4,',  EXCEEDS STORAGE=',I4)
 2007  FORMAT(1H0,'NO TABLE FOUND FOR ID1, ID2=', 2I10)
 2008  FORMAT(1H0,'NO. OF JAMMERS=',I4,',  EXCEEDS STORAGE=',I4)
 2009  FORMAT(1H0,'NO. OF SONARS=',I4,',  EXCEEDS STORAGE=',I4)
 2011  FORMAT(1H0,'AT END OF SETUPE, CUM NO. SETUP ERRORS=NFLAG=',I5)
 3000  FORMAT(7X,'BLU PLATFORMS VS. SYSTEMS')
 3001  FORMAT(7X,'RED PLATFORMS VS. SYSTEMS')
 3002  FORMAT(9X,'SYSTEM',13X, 8(I3,'.',2A4) )
 3003  FORMAT(9X,'NAME AND TYPE',6X, 8I12)
 3004  FORMAT(5X,I3,'.',2A4,I9,2X, 8I12)
 3006  FORMAT(6X,'SONAR DATA'/
     1    8X,'NO.BLU SONAR TYPES=',I4,',  NO.RED SONAR TYPES=',I4)
 3008  FORMAT(/1X,I6,'.',2A4,I9,'     SONAR DATA')
 3010  FORMAT(8X,10G12.5)
 6700 FORMAT(/6X,3A4,'RADAR ARRAY DATA'/6X,10('  COL.=',I3,2X))
 6701 FORMAT( 1X,I4,1X,10G12.5)
 6702 FORMAT(/6X,A4,' JAMMER ARRAY DATA'/)
 6703 FORMAT( 1X,I4,1X,3G12.5,5X,3G12.5)
       KZARL = 96 + LA*(5 + MA)
       JZARL = KZARL
       NFLAG2= 0
       NSONMX = 15
       LASON  = 12
       NSON   = 0
       NBSON  = 0
       NRSON  = 0
C***  SETUP BLU RADARS
 1     IF (KBU.LE.0)  GO TO 201
C***  CYCLE THRU BLU CATALOG RADAR TABLES
       IY   = 3
       NBTR = 0
       NBSR = 0
       DO 46 KC=1,NBRTT
       ID1 = KBRTT(KC)
      ICTPE=ID1/1000000
       DO 44 KX=1,3
       ID2 = -KX
      CALL   RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF ( IXX.EQ.0)  GO TO 26
      WRITE(N6,2007) ID1,ID2
      GO TO 44
C***  CYCLE THRU SYSTEM TYPES IN  KBST (FROM ENGAGEMENT STRUCTURE)
   26 DO 42 KT=1,NBSS
      K    = KBST(KT) / 10000
     IF(K/100 .NE. IDTPE) GO TO 42
      DO 28 J=1,LA1
      IF (K.EQ.KA1( J,5))  GO TO 30
 28   CONTINUE
      GO TO 42
C***  FOUND, SYSTEM TYPE IN CATALOG
 30   JT   = J
      IF (KX.GE.2)  GO TO 37
C***  TRANSFER RADAR  NAME AND DATA. PUT POINTER IN  KBST.
      IF (IDTPE.NE.65)  GO TO 32
      NBSR = NBSR + 1
      JR    = NBSR
      GO TO 34
   32 NBTR = NBTR + 1
      JR    = NBTR + NBSR
 34   IF (JR.LE.NBRMX)  GO TO 36
      WRITE(N6,2006) JR, NBRMX
      NFLAG2 = NFLAG2 + 1
      GO TO 48
 36   NABSS(1,KT) = KA1(JT,2)
      NABSS(2,KT) = KA1(JT,3)
      KBST(KT) =  KBST(KT) + JR
      GO TO 38
 37   JR   = MOD(KBST(KT),100)
      IF (JR.GT.0)  GO TO 38
      E1   = 37.01
      WRITE(N6,2005) E1, ID1, ID2, NBSR, NBTR
      NFLAG2 = NFLAG2 + 1
      GO TO 42
 38   JA   = 12*(KX-1)
      DO 40 J=1,12
 40   BRAD(JA+J,JR) = VA1(JT,J)
 42   CONTINUE
 44   CONTINUE
 46   CONTINUE
C***  SET UP BLU JAMMERS
 48   IY   = 3
      ID1 = 68100000
      ID2 = -1
      CALL   RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF ( IXX.EQ.0)  GO TO 50
      WRITE(N6,2007) ID1,ID2
      GO TO 59
C***  CYCLE THRU SYSTEM TYPES IN KBST
 50   NBJ = 0
      DO 58 KT=1,NBSS
      K    = KBST(KT) / 10000
      DO 52 J= 1,LA1
      IF (K.EQ.KA1( J,5))  GO TO 54
 52   CONTINUE
      GO TO 58
C***  FOUND JAMMER  IN CATALOG
 54   JT  = J
      NBJ = NBJ + 1
      IF (NBJ.LE.NBJMX)  GO TO 56
      WRITE(N6,2008) NBJ, NBJMX
      NFLAG2 = NFLAG2 + 1
      NBJ = NBJMX
      GO TO 59
C***  TRANSFER JAMMER NAME AND DATA. PUT POINTER IN KBST.
C***  NOTE. FREQ. = 30000 / XLAMDA (IN CM.)  (FREQ.IN MHZ)
 56   NABSS(1,KT) = KA1(JT,2)
      NABSS(2,KT) = KA1(JT,3)
      KBST(KT) =  KBST(KT) + NBJ
      DO 57 J=1,3
      I   = 4*(J - 1)
      BJAM(1,J,NBJ) = VA1(JT,I+3)
      BJAM(2,J,NBJ) = VA1(JT,I+4)
      BJAM(3,J,NBJ) = 1.E6*(VA1(JT,I+2)-VA1(JT,I+1) )
      BJAM(4,J,NBJ) = 1.0
      BJAM(5,J,NBJ) = VA1(JT,I+1)
 57   BJAM(6,J,NBJ) = VA1(JT,I+2)
 58   CONTINUE
C*** SETUP BLU SONARS
 59   IY  = 3
      ID1 = 65300000
      ID2 = -1
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX.EQ.0)  GO TO  60
      WRITE(N6,2007)  ID1,ID2
      GO TO  68
C*** CYCLE THRU SYSTEM TYPES IN KBST
 60   DO   66 KT=1,NBSS
      K    = KBST(KT) / 10000
      IF ( K / 100.NE. 65)  GO TO  66
      DO   61 J= 1,LA1
      IF (K.EQ.KA1( J,5))  GO TO  62
 61   CONTINUE
      GO TO  66
C*** FOUND SONAR IN CATALOG
 62   JT  = J
      NSON = NSON + 1
      IF (NSON.LE.NSONMX)  GO TO  63
      WRITE(N6,2009) NSON, NSONMX
      NFLAG2 = NFLAG2 + 1
      NSON = NSONMX
      GO TO  68
C*** TRANSFER SONAR NAME AND DATA.  PUT POINTER.
 63   DO  64 I=1,2
      KNAME = KA1(JT,I+1)
      NABSS(I,KT) = KNAME
 64   NAMSON(I,NSON) = KNAME
      KSONTY(NSON) = K*10000
```

```
       KBST(KT) = KBST(KT) + NSON
       DO  65 I=1,12
   65  SONSYS(I, NSON) = VA1(JT,I)
   66  CONTINUE
   68  NBSON = NSON
C***  PRINT OPTION
       IF (IPRINT.LT.1)  GO TO 73
       KC  = 1
  70   KD  = KC + 7
       KD  = MIN0(KD,NBP)
       CALL  PAGE
       WRITE(N6,3000)
       WRITE(N6,3002) ((J,  NAMBP(1,J),NAMBP(2,J)),J=KC,KD)
       WRITE(N6,3003) ( KBPT(J),J=KC,KD)
       DO 72 I=1,NBSS
       WRITE(N6,3004) (I,NABSS(1,I),NABSS(2,I),KBST(I),
     1   (KBPXS(I,J),J=KC,KD) )
  72   CONTINUE
       KC  = KC + 8
       IF (KC.LE.NBP)  GO TO 70
  73   IF (IPRINT.LT.2)  GO TO 201
      IF(NBSR.LE.0) GO TO 78
      NA=1
      NB=NBSR
      IC=1
      IT=1
   75 CALL PAGE
      WRITE(N6,6700) PCOL(IC),(PRTPE(I,IT),I=1,2),(J,J=NA,NB)
      DO  76 I=1,36
  76  WRITE(N6,6701) I,(BRAD(I,J),J=NA,NB)
      IF(IT.EQ.2) GO TO 80
  78  IF(NBTR.LE.0) GO TO 80
      IT=2
      NA=NBSR+1
      NB=NBSR+NBTR
      GO TO 75
  80  CALL PAGE
      WRITE(N6,6702) PCOL(1)
      DO  82 I=1,6
  82  WRITE(N6,6703) I, ((BJAM(I,J,K),J=1,3),K=1,2)
C***  SETUP RED RADARS
 201   IF (KRU.LE.0)  GO TO 590
C***  CYCLE THRU RED CATALOG RADAR TABLES
       IY  = 3
       NRTR = 0
       NRSR = 0
       DO 246 KC=1,NRRTT
       ID1 = KRRTT(KC)
      IDTPE=ID1/1000000
       DO 244 KX=1,3
       ID2 = -KX
       CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
       IF (IXX.EQ.0)  GO TO 226
       WRITE(N6,2007) ID1,ID2
       GO TO 244
```

```
C***  CYCLE THRU SYSTEM TYPES IN  KRST (FROM ENGAGEMENT STRUCTURE)      SETE2650
 226   DO 242 KT=1,NRSS                                                 SETE2660
       K    = KRST(KT) / 10000                                          SETE2670
      IF(K/100 .NE. IDTPE) GO TO 242                                    SETE2680
       DO 228 J=1,LA1                                                   SETE2690
       IF (K.EQ.KA1( J,5))  GO TO 230                                   SETE2700
 228   CONTINUE                                                         SETE2710
       GO TO 242                                                        SETE2720
C***  FOUND, SYSTEM TYPE IN CATALOG                                     SETE2730
 230   JT   = J                                                         SETE2740
       IF (KX.GE.2)  GO TO 237                                          SETE2750
C***  TRANSFER RADAR  NAME AND DATA. PUT POINTER IN  KRST.              SETE2760
       IF (IDTPE.NE.85)  GO TO 232                                      SETE2770
       NRSR = NRSR + 1                                                  SETE2780
       JR    = NRSR                                                     SETE2790
       GO TO 234                                                        SETE2800
 232   NRTR = NRTR + 1                                                  SETE2810
       JR    = NRTR + NRSR                                              SETE2820
 234   IF (JR.LE.NRRMX)  GO TO 236                                      SETE2830
       WRITE(N6,2006) JR, NRRMX                                         SETE2840
       NFLAG2 = NFLAG2 + 1                                              SETE2850
       GO TO 248                                                        SETE2860
 236   NARSS(1,KT) = KA1(JT,2)                                          SETE2870
       NARSS(2,KT) = KA1(JT,3)                                          SETE2880
       KRST(KT) =   KRST(KT) + JR                                       SETE2890
       GO TO 238                                                        SETE2900
 237   JR   = MOD(KRST(KT),100)                                         SETE2910
       IF (JR.GT.0)  GO TO 238                                          SETE2920
       E1   = 237.01                                                    SETE2930
       WRITE(N6,2005) E1, ID1, ID2, NRSR, NRTR                          SETE2940
       NFLAG2 = NFLAG2 + 1                                              SETE2950
       GO TO 242                                                        SETE2960
 238   JA   = 12*(KX-1)                                                 SETE2970
       DO 240 J=1,12                                                    SETE2980
 240   RRAD(JA+J,JR) = VA1(JT,J)                                        SETE2990
 242   CONTINUE                                                         SETE3000
 244   CONTINUE                                                         SETE3010
 246   CONTINUE                                                         SETE3020
C***  SET UP RED JAMMERS                                                SETE3030
 248   IY   = 3                                                         SETE3040
       ID1 = 8100000                                                    SETE3050
       ID2 = -1                                                         SETE3060
       CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                 SETE3070
       IF ( IXX.EQ.0)  GO TO 250                                        SETE3080
       WRITE(N6,2007) ID1,ID2                                           SETE3090
       GO TO 259                                                        SETE3100
C***  CYCLE THRU SYSTEM TYPES IN KRST                                   SETE3110
 250   NRJ = 0                                                          SETE3120
       DO 258 KT=1,NRSS                                                 SETE3130
       K    = KRST(KT) / 10000                                          SETE3140
       DO 252 J= 1,LA1                                                  SETE3150
       IF (K.EQ.KA1( J,5))  GO TO 254                                   SETE3160
 252   CONTINUE                                                         SETE3170
       GO TO 258                                                        SETE3180
C***  FOUND JAMMER IN CATALOG                                           SETE3190
```

```
 254    JT  = J
        NRJ = NRJ + 1
        IF (NRJ.LE.NRJMX)  GO TO 256
        WRITE(N6,2008) NRJ, NRJMX
        NFLAG2 = NFLAG2 + 1
        NRJ = NRJMX
        GO TO 259
C***  TRANSFER JAMMER NAME AND DATA. PUT POINTER IN KRST.
C***  NOTE. FREQ. = 30000 / XLAMDA (IN CM.)   (FREQ.IN MHZ)
 256    NARSS(1,KT) = KA1(JT,2)
        NARSS(2,KT) = KA1(JT,3)
        KRST(KT) =  KRST(KT) + NRJ
        DO 257 J=1,3
        I   = 4*(J - 1)
        RJAM(1,J,NRJ) = VA1(JT,I+3)
        RJAM(2,J,NRJ) = VA1(JT,I+4)
        RJAM(3,J,NRJ) = 1.E6*(VA1(JT,I+2)-VA1(JT,I+1) )
        RJAM(4,J,NRJ) = 1.0
        RJAM(5,J,NRJ) = VA1(JT,I+1)
 257    RJAM(6,J,NRJ) = VA1(JT,I+2)
 258    CONTINUE
C*** SETUP REC SONARS
 259    IY  = 3
        ID1 = 85300000
        ID2 = -1
        CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
        IF ( IXX.EQ.0)  GO TO 260
        WRITE(N6,2007)  ID1,ID2
        GO TO 268
C*** CYCLE THRU SYSTEM TYPES IN KRST
 260    DO 266 KT=1,NRSS
        K   = KRST(KT) / 10000
        IF ( K / 100.NE. 85)  GO TO 266
        DO 261 J= 1,LA1
        IF (K.EQ.KA1( J,5))  GO TO 262
 261    CONTINUE
        GO TO 266
C*** FOUND SONAR IN CATALOG
 262    JT  = J
        NSON = NSON + 1
        IF (NSON.LE.NSONMX)  GO TO 263
        WRITE(N6,2009) NSON, NSONMX
        NFLAG2 = NFLAG2 + 1
        NSON = NSONMX
        GO TO 268
C*** TRANSFER SONAR NAME AND DATA.  PUT POINTER.
 263    DO 264 I=1,2
        KNAME = KA1(JT,I+1)
        NARSS( I,KT) = KNAME
 264    NAMSON(I,NSON) = KNAME
        KSONTY(NSON) = K*10000
        KRST(KT) = KRST(KT) + NSON
        DO 265 I=1,12
 265    SONSYS(I, NSON) = VA1(JT,I)
 266    CONTINUE
 268    NRSON = NSON - NBSON                                                      SETE3750
C***   PRINT OPTION                                                              SETE3760
        IF (IPRINT.LT.1)  GO TO 273                                              SETE3770
        KC   = 1                                                                 SETE3780
 270    KD   = KC + 7                                                            SETE3790
        KD   = MIN0(KD,NRP)                                                      SETE3800
        CALL  PAGE                                                               SETE3810
        WRITE(N6,3001)                                                           SETE3820
        WRITE(N6,3002) ((J,   NAMRP(1,J),NAMRP(2,J)),J=KC,KD)                    SETE3830
        WRITE(N6,3003) ( KRPT(J),J=KC,KD)                                        SETE3840
        DO 272 I=1,NRSS                                                          SETE3850
        WRITE(N6,3004) (I,NARSS(1,I),NARSS(2,I),KRST(I),                         SETE3860
     1   (KRPXS(I,J),J=KC,KD) )                                                  SETE3870
 272    CONTINUE                                                                 SETE3880
        KC   = KC + 8                                                            SETE3890
        IF (KC.LE.NRP)  GO TO 270                                                SETE3900
 273    IF (IPRINT.LT.2)  GO TO 590                                              SETE3910
       IF(NRSR.LE.0) GO TO 278                                                   SETE3920
       NA=1                                                                      SETE3930
       NB=NRSR                                                                   SETE3940
        IC=2                                                                     SETE3950
       IT=1                                                                      SETE3960
  275  CALL PAGE                                                                 SETE3970
       WRITE(N6,6700) PCOL(IC),(PRTPE(I,IT),I=1,2),(J,J=NA,NB)                   SETE3980
       DO 276 I=1,36                                                             SETE3990
 276   WRITE(N6,6701) I,(RRAD(I,J),J=NA,NB)                                      SETE4000
       IF(IT.EQ.2) GO TO 280                                                     SETE4010
 278   IF(NRTP.LE.0) GO TO 280                                                   SETE4020
       IT=2                                                                      SETE4030
       NA=NRSR+1                                                                 SETE4040
       NB=NRSR+NRTP                                                              SETE4050
       GO TO 275                                                                 SETE4060
 280   CALL PAGE                                                                 SETE4070
       WRITE(N6,6702) PCOL(2)                                                    SETE4080
       DO 282 I=1,6                                                              SETE4090
 282   WRITE(N6,6703) I, ((RJAM(I,J,K),J=1,3),K=1,2)                             SETE4100
C*** PRINT SONAR DATA                                                            SETE4110
 301    CALL PAGE                                                                SETE4120
        WRITE(N6,3006)  NBSON, NRSON                                             SETE4130
        DO 302 I=1,NSON                                                          SETE4140
        WRITE(N6,3008) I, (NAMSON(J,I),J=1,2), KSCNTY(I)                         SETE4150
 302    WRITE(N6,3010) (SONSYS(J,I),J=1,LASON)                                   SETE4160
 590    IF (NFLAG2.NE.0)  NFLAG = NFLAG + 1                                      SETE4170
        WRITE(N6,2011) NFLAG                                                     SETE4180
        RETURN                                                                   SETE4190
        END                                                                      SETE4200


        SUBROUTINE  SETUPF(IPRINT)                                               SETF0010
C   PGM=NXX(NEM). L.D.G.         VER.2     11-28-73     FORTRAN IV      EBCD     SETF0020
C     TO SET UP GENERAL DATA AND CONTROL                                         SETF0030
C       1.TABLE 3110 SHIP VULN.                                                  SETF0040
C                                                                                SETF0050
C   CALLED BY MAIN AND ZIP CARD AS FOLLOWS                                       SETF0060
```

```
CIP  8 6 P                                      SET UP MISCELLANEOUS DATA       SETF0070
C***                                                                            SETF0080
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12                     SETF0090
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                              SETF0100
     1,IDUMB(72), NFLAG, NFLAG2                                                 SETF0110
CZAZA                                                                           SETF0120
      INTEGER TITLA                                                             SETF0130
      COMMON/CZAZA/NCBA,IDA1,IDA2,IDA3,JZARL,LZA,MZA,ICA(16),CA(16)             SETF0140
     1, TITLA(15),HEADA(37),IFMT                                                SETF0150
     2, LA1,LAA1,MA1,MAA1,KA1(27,5),VA1(27,12)                                  SETF0160
      DIMENSION  BZA( 555)                                                      SETF0170
      EQUIVALENCE  (NCBA,BZA(1))                                                SETF0180
      DATA  NCB,LZ,MZ,LA,MA/ 1,555,12,27,12/                                    SETF0190
CNAVIG                                                                          SETF0200
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                          SETF0210
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18 ),NBU(18),             SETF0220
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18 ),NRU(18),             SETF0230
     3          TTIME,NUMX,                                                     SETF0240
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),           SETF0250
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)            SETF0260
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                             SETF0270
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),                   SETF0280
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),                 SETF0290
     2  BPX1(12,15),                                                            SETF0300
     3               NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),                 SETF0310
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),                 SETF0320
     5  RPX1(12,15)                                                             SETF0330
CMSLSYS                          16     2     2     6     6     24     50       SETF0340
      COMMON/CMSLSY/ NMSL,NMSLMX,NBSSM,NBASM,NRSSM,NRASM,LATMSL,LTRAJ,          SETF0350
     1     NAMMSL(2,16),KMSLTY(16),                                             SETF0360
     2     DATMSL(24,16), TTRAJ(50,16)                                          SETF0370
CWORK                                                                           SETF0380
      COMMON/CWORK/                                                             SETF0390
     1     LW1,MW1,LAW1,MAW1,  KW1(27,5),  VW1(27,12),                          SETF0400
     2     LW2,MW2,LAW2,MAW2,  KW2(27,5),  VW2(27,12),                          SETF0410
     3     LW3,MW3,LAW3,MAW3,  KW3(27,5),  VW3(27,12),         PADW(111)        SETF0420
CZZSYS                           25      11      12   15                        SETF0430
      COMMON/CZZSYS/NZZSYS,NZZSMX,NBZSYS,NRZSYS,LAZZ,NAMSYS(2,25),              SETF0440
     1     KZZTYP(25),ZZSYS(15,25)                                              SETF0450
CSAMLT               8,12,15,                                                   SETF0460
      COMMON/CSAMLT/L,M,N,RMN(15),RMX(15),EMN(15),EMX(15),ZMX(15),              SETF0470
     1 XMX(15),  RT(8,12,15)                                                    SETF0480
CSASYS                            15     4     4     2     4    20              SETF0490
      COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,               SETF0500
     1     NAMSA(2,15),KSATYP(15),SASYS(20,15)                                  SETF0510
CRADAR                                                                          SETF0520
      COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                                  SETF0530
     1  BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,3,2),                  SETF0540
     2                NRSR,NRTR,NRRMX,NRJ,NRJMX,                                SETF0550
     3  RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,3,2)                   SETF0560
      COMMON/INOUU/ IPR(16),JPAR(16), PAR(16),LABEL                             SETF0570
CKILLF                                                                          SETF0580
      COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,                 SETF0590
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULCST(100),               SETF0600
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                                SETF0610
```

```
CHOMER                                                                          SETF0620
      COMMON/CHOMER/ NHOM,NHOMMX,NAMHOM(2,10),KHOMTY(10),DATHOM(36,10)        SETF0630
CFORMATS                                                                        SETF0640
 2006  FORMAT(1H0,'NO.HOMING RADARS=',I4,', EXCEEDS STORAGE=',I4/)              SETF0650
 2007  FORMAT(1H0,'NO TABLE FOUND FOR ID1, ID2=', 2I10)                         SETF0660
 2008  FORMAT(1H0,'DATA MISSING FOR HOMING RADAR TYPE=',I9/)                    SETF0670
 2011  FORMAT(1H0,'AT END OF SETUPF, CUM NO. SETUP ERRORS=NFLAG=',I5)           SETF0680
 2012  FORMAT(6X,'NO.OF CRUISE MISSILE HOMING RADAR TYPES=',I4)                 SETF0690
 2013  FORMAT(/1X, I6,'.',2A4,I9,4X,'HOMING RADAR DATA')                        SETF0700
 2014  FORMAT(8X,10G12.5)                                                       SETF0710
       KZARL = 96 + LA*(5 + MA)                                                 SETF0720
       JZARL = KZARL                                                            SETF0730
       NFLAG2 = 0                                                               SETF0740
       LVSHP  = 5                                                               SETF0750
       MVSHP  = 12                                                              SETF0760
       NVSHP  = 1                                                               SETF0770
 100    IY = 3                                                                  SETF0780
       KTRAN = 1                                                                SETF0790
       ID1 = 31100000                                                           SETF0800
       ID2 = -1                                                                 SETF0810
C*** READ FROM DISK 11                                                          SETF0820
 102   CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)                         SETF0830
       IF (IXX .EQ. 0)   GO TO (104,104,104), KTRAN                             SETF0840
       WRITE(N6,2007) ID1,ID2                                                   SETF0850
       NFLAG2 = NFLAG2 + 1                                                      SETF0860
       GO TO (110,900,900), KTRAN                                               SETF0870
 104   DO 106 I=1,LVSHP                                                         SETF0880
       DO 106 J=1,MVSHP                                                         SETF0890
 106   VULSHP(I,J,1) = VA1(I,J)                                                 SETF0900
       IF (IPRINT .LT. 2)  GO TO 108                                            SETF0910
       CALL PAGE                                                                SETF0920
       CALL  WMAT3(2,ICA,VULSHP,LVSHP,MVSHP,NVSHP,LVSHP,MVSHP, 1,NLINE,         SETF0930
     1 60HVULNERABILITY DATA, NONNUC, SHIPS                              )      SETF0940
 108   CONTINUE                                                                 SETF0950
C***   SETUP CRUISE MISSILE HOMING RADARS                                       SETF0960
 110   NHOMMX = 10                                                              SETF0970
       NHOM   =  0                                                              SETF0980
       IF (NMSL .LE. 0)   GO TO 900                                             SETF0990
       KH = 0                                                                   SETF1000
       DO 112 MT=1,NMSL                                                         SETF1010
       KHT = DATMSL(11,MT) + .1                                                 SETF1020
       IF (KHT/10.NE.874)  GO TO 112                                            SETF1030
       KH = KH + 1                                                              SETF1040
       KHOMTY(KH) = KHT * 10000                                                 SETF1050
 112   CONTINUE                                                                 SETF1060
       NHOM = KH                                                                SETF1070
       IF (KH.GT.0)   GO TO 114                                                 SETF1080
       NHOM = 0                                                                 SETF1090
       GO TO 142                                                                SETF1100
 114   CALL  KORDER(KHOMTY,NHOM)                                                SETF1110
       CALL  UNIQUE(KHOMTY,NHOM,K)                                              SETF1120
       NHOM = K                                                                 SETF1130
       IF (NHOM.LE.NHOMMX)   GO TO 116                                          SETF1140
       WRITE(N6,2006) NHOM, NHOMMX                                              SETF1150
       NHOM = NHOMMX                                                            SETF1160
      NFLAG2 = NFLAG2 + 1
 116  DO 118 I=1,36
      DO 118 J=1,NHOMMX
 118  DATHOM(I,J) = .0
      IY  = 3
      ID1 = 87100000
      DO 132 KX=1,3
      ID2 = - KX
      CALL  RCBD11(BZA,LZ,MZ,NCB,KZARL,ID1,ID2,IY,IXX)
      IF (IXX.EQ.0)    GO TO 120
      WRITE(N6,2007) ID1,ID2
      NFLAG2 = NFLAG2 + 1
      NHOM = 0
      GO TO 142
C***  MATCH & TRANSFER
 120  DO 130 KH=1,NHOM
      KHT = KHOMTY(KH) / 10000
      DO 122 KR=1,LA1
      KRT = KA1(KR,5)
      IF (KHT .EQ. KRT)   GO TO 124
 122  CONTINUE
      WRITE(N6,2008) KHT
      KHOMTY(KH) = -1
      NFLAG2 = NFLAG2 + 1
      GO TO 130
 124  KL   = KR
      IF (KX .GT. 1)   GO TO 126
      NAMHOM(1,KH) = KA1(KL,2)
      NAMHOM(2,KH) = KA1(KL,3)
 126  JA   = 12*(KX - 1)
      DO 128 J=1,12
 128  DATHOM(JA+J,KH) = VA1(KL,J)
 130  CONTINUE
 132  CONTINUE
C***  BACKFILL POINTERS IN CRUISE MISSILE = KRST(   )
      DO 140 KS=1,NRSS
      KST = KRST(KS)
      KTEST = KST / 1000000
      IF (KTEST.NE.96 .AND. KTEST.NE.98)   GO TO 140
      MP   = MOD(KST,100)
      KHTR = DATMSL(11,MP) + .1
      DO 136 KH=1,NHOM
      KHT  = KHOMTY(KH) / 10000
      IF (KHT .EQ. KHTR)   GO TO 138
 136  CONTINUE
      GO TO 140
 138  KRST(KS) = KRST(KS) + KH*100
 140  CONTINUE
C***  PRINT OPTIONS
 142  IF (IPRINT.LT.1)    GO TO 150
      CALL  PAGE
      WRITE(N6,2012) NHOM
      IF (NHOM.LE.0)      GO TO 150
      DO 144 I=1,NHOM
      WRITE(N6,2013) I, (NAMHOM(K,I),K=1,2), KHOMTY(I)
      IF (IPRINT.LT.2)   GO TO 144
      WRITE(N6,2014) (DATHOM(K,I),K=1,36)
  144 CONTINUE
  150 CONTINUE
C***
  900 IF (NFLAG2.NE.0)  NFLAG=NFLAG+1
      WRITE(N6,2011) NFLAG
      RETURN
      END


      SUBROUTINE  UCHEK(KODE,N5,N6)
C  NEM UTILITY CHECKER.          5 NOV 73
C  KODE IS ZIP CODE, USED TO CALL UCHEK AS FOLLOWS
CIP  6 A B                            CALL UCHEK
C       A  = 1, NAVIG CHEK       B = 1, NO READ NUCHEK.  = 2, READ.
C       A  = 2, RADAR CHEK       B = 1, ONE RADAR. SINGLE TARGET
C                                B = 2, RUN ENGAGEMENT LIST. SINGLE TARGET
C***
      DIMENSION  KODE(19),TIMES(10)
      DATA  NTIMES,TIMES/  2, -2., -1., 8*0./
      NAMELIST/NUCHEK/NTIMES,TIMES
     K2=KODE(2)
     K3=KODE(3)
     K4=KODE(4)
    1 GO TO (   2, 52, 52), K3
C***  NAVIG CHEK
    2 IF (KODE(4).LE.1)   GO TO 6
      READ (N5,NUCHEK)
    6 WRITE(N6,NUCHEK)
      IPRINT = 1
      DO 10 I=1,NTIMES
      TIME = TIMES(I)
     *CALL   ALLXYZ(TIME,IPRINT,N6,IXX)
   10 CONTINUE
      RETURN
C***  RADAR CHEK
   52 CALL  RADCK(K4,N5,N6)
      RETURN
      END


      SUBROUTINE RADCK(K4,N5,N6)
C  NEM RADAR CHECKER             5 NOV 73
C    K4 = 1, SINGLE RADAR
C    K4 = 2, RUN ENGAG.LIST.  SINGLE TARGET
C    K4 = 3,
CPLAT  VS  SYSTEMS  (SUBSYSTEMS)
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),
     2  BPX1(12,15),
     3                NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),
     5   RPX1(12,15)
CRADAR
       COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,
     1   BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,6),
     2                 NRSR,NRTR,NRRMX,NRJ,NRJMX,
     3   RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,6)
       DIMENSION RANGE(10),TRAD(10),PD(10),ANG(10),DYN(10)
       NAMELIST /NAM1/ IBR,KTYP,IRNG,TRAD,RANGE,BENV,RENV,BETC,RETC,
     1 BTAR,RTAR,BJAM,RJAM, IPRINT,ISTOP
       NAMELIST/NAM2/ IBR,KTYP,IRNG,IPRINT,ISTOP
       NAMELIST/NAM3/ KTYP,IRNG
       ISTOP = 0
       TRAD(1) = 0.
 1     ITST    = 0
       CALL PAGE
       READ (N5,NAM1)
       WRITE(N6,NAM2)
C      GO TO (  4,    ,    ), K4
 4     IPRINT = 2
       GO TO ( 10, 20), IBR
C** SINGLE RADAR
 10   RHOR = 1.2289*(SQRT(BETC(1,1)) + SQRT(RTAR(3,1)) )
      IF (RANGE(1).GT.RHOR)  RANGE(1)=RHOR
       DO   12 I=1,NBSS
       K = KBST(I) / 10000
       IF (KTYP .EQ. K)  GO TO 16
 12    CONTINUE
 14    IF (ISTOP.GE.1)  RETURN
       GO TO 1
 16    K = MOD(KBST(I),100)
       WRITE(N6,NAM3)
       NMODE = BRAD(36,K) + .1
       BETC(4,1) = NMODE
       DELR = RTAR(4,1) / BRAD(21,K)
       IF (NMODE.EQ.2)  DELR= RTAR(4,1) / 60.
 17    CALL RADAR(BRAD(1,K),BENV(1,1),BETC(1,1),RTAR(1,1),RJAM(1,1),
     1 TRAD,RANGE,PD,ANG,DYN,ITST,IRNG,IPRINT)
       IF (RANGE(1).LE.DELR + 2.)  GO TO 14
       RANGE(1) = RANGE(1) - DELR
       IPRINT = 1
       TRAD(1) = TRAD(10) + TRAD(2) - TRAD(1)
       GO TO 17
 20   RHOR = 1.2289*(SQRT(RETC(1,1)) + SQRT(BTAR(3,1)) )
      IF (RANGE(1).GT.RHOR)  RANGE(1)=RHOR
       DO   22 I=1,NRSS
       K = KRST(I) / 10000
       IF (KTYP .EQ. K)  GO TO 26
 22    CONTINUE
       GO TO 14
 26    K = MOD(KRST(I),100)
       WRITE(N6,NAM3)
       NMODE = RRAD(36,K) + .1
       RETC(4,1) = NMODE
       DELR = BTAR(4,1) / RRAD(21,K)
       IF (NMODE.EQ.2)  DELR= BTAR(4,1) / 60.
 27     CALL RADAR(RRAD(1,K),RENV(1,1),RETC(1,1),BTAR(1,1),BJAM(1,1),      RADK0670
      1 TRAD,RANGE,PD,ANG,DYN,ITST,IRNG,IPRINT)                            RADK0680
        IF (RANGE(1).LE.DELR + 2.)  GO TO 14                               RADK0690
        RANGE(1) = RANGE(1) - DELR                                         RADK0700
        IPRINT = 1                                                         RADK0710
        TRAD(1) = TRAD(10) + TRAD(2) - TRAD(1)                             RADK0720
        GO TO 27                                                           RADK0730
        END                                                                RADK0740


      SUBROUTINE XECUTE                                                    XECU0010
CNAVIG                                                                     XECU0020
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                     XECU0030
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),         XECU0040
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),         XECU0050
     3          TTIME,NUMX,                                                XECU0060
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),      XECU0070
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)       XECU0080
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12                XECU0090
      COMMON/STRCON/  JPK,IPK                                              XECU0100
      COMMON/ INFO/  LENIFO,LENSEG,MAXSEG,NEXSEG,LSTSEG,LZCSEG             XECU0110
      COMMON/EXSTOR/ LENSYS,MAXRTU,MAXBTU,LEFTDS,NRTU,NBTU,IBRS,NPHASE     XECU0120
C                                                                          XECU0130
C-----THE FOLLOWING TWO STATEMENTS ADJUST THE AMOUNT OF STORAGE             XECU0140
C     AVAILABLE TO THE PROGRAM FOR DYNAMIC ALLOCATION.                     XECU0150
C                                                                          XECU0160
      DIMENSION STORE1(15000),ISTOR1(15000)                                XECU0170
      DATA LDIMEN/15000/                                                   XECU0180
      EQUIVALENCE (STORE1(1),ISTOR1(1))                                    XECU0190
C                                                                          XECU0200
C-----THE MINIMUM DIMENSION=LDIMEN, FOR ARRAYS STORE1 AND ISTOR1           XECU0210
C     IS GIVEN BY---                                                       XECU0220
C                                                                          XECU0230
C        LDIMEN=(KBU+KRU)*(NSYSPU+8*NSEGPU + 5)+13*(MAXBTU+MAXRTU)+128     XECU0240
C                                                                          XECU0250
      DIMENSION NAME(2)                                                    XECU0260
      DATA NAME/'STOR','E1  '/                                             XECU0270
      DATA KBUSEG,KRUSEG,ITRSEG,ITBSEG,INFORM/5*0/                         XECU0280
      DATA LBRSYS,LBSYST,LRSYST/3*0/                                       XECU0290
      DATA IRTU,IBTU,IRUXYZ,IBUXYZ,INVRTU,INVBTU,IBSTAT,IRSTAT/8*0/        XECU0300
      NAMELIST /DIMENS/ LENSEG,NSEGPU,NSYSPU,MAXRTU,MAXBTU,                XECU0310
     1     LDIMEN,KSAVE,NPHASE                                             XECU0320
      KSAVE=0                                                              XECU0330
      NPHASE=0                                                             XECU0340
      IPK=10000                                                            XECU0350
      JPK=1000*IPK                                                         XECU0360
      DO 100 I=1,LDIMEN                                                    XECU0370
  100 ISTOR1(I)=0                                                          XECU0380
      ISTOR1(1)=NAME(1)                                                    XECU0390
      ISTOR1(2)=NAME(2)                                                    XECU0400
      ISTOR1(3)=LDIMEN                                                     XECU0410
      ISTOR1(4)=10                                                         XECU0420
      ISTOR1(5)=LDIMEN                                                     XECU0430
      ISTOR1(6)=0                                                          XECU0440
      ISTOR1(7)=10*IPK+5                                                XECU0450
      MAXBTU=KBU                                                        XECU0460
      MAXRTU=KRU                                                        XECU0470
      LENSEG=8                                                          XECU0480
      NSYSPU=10                                                         XECU0490
C----   INPUT STORAGE ALLOCATION PARAMETERS                             XECU0500
      CALL PAGE                                                         XECU0510
      READ(N5,DIMENS)                                                   XECU0520
C                                                                       XECU0530
      JRU=KRU+MAXRTU                                                    XECU0540
      JBU=KBU+MAXBTU                                                    XECU0550
      IBU=2*KBU+11                                                      XECU0560
      IRU=2*KRU+11                                                      XECU0570
      MAXBT2=2*MAXBTU                                                   XECU0580
      MAXRT2=2*MAXRTU                                                   XECU0590
      LENSYS=(KRU+KBU)*NSYSPU                                           XECU0600
      LENIFO=LDIMEN-13*(MAXBTU+MAXRTU)-(NSYSPU+5)*(KBU+KRU)-128         XECU0610
      IF(LENIFO.GT.9999) LENIFO=9999                                    XECU0620
      IF(LENIFO.LT.1) LENIFO=1                                          XECU0630
      NSEGPU=LENIFO/(LENSEG*(KRU+KBU))                                  XECU0640
      WRITE(N6,DIMENS)                                                  XECU0650
      MAXSEG=LENIFO-LENSEG+1                                            XECU0660
      NEXSEG=0                                                          XECU0670
      LSTSEG=1-LENSEG                                                   XECU0680
      IF(KBUSEG.LE.0) CALL SETDS(KBUSEG,ISTOR1,'KBUSEG  ',0,KBU,  0,0)  XECU0690
      IF(KRUSEG.LE.0) CALL SETDS(KRUSEG,ISTOR1,'KRUSEG  ',0,KRU,  0,0)  XECU0700
      IF(ITRSEG.LE.0) CALL SETDS(ITRSEG,ISTOR1,'ITRSEG  ',0,JRU,0,0)    XECU0710
      IF(ITBSEG.LE.0) CALL SETDS(ITBSEG,ISTOR1,'ITBSEG  ',0,JBU,0,0)    XECU0720
      IF(LBSYST.LE.0) CALL SETDS(LBSYST,ISTOR1,'LBSYST  ',0,IBU,0,0)    XECU0730
      IF(LRSYST.LE.0) CALL SETDS(LRSYST,ISTOR1,'LRSYST  ',0,IRU,0,0)    XECU0740
      IF(IRTU.LE.0) CALL SETDS(IRTU,ISTOR1,'IRTU    ',0,MAXRT2,0,0)     XECU0750
      IF(IBTU.LE.0) CALL SETDS(IBTU,ISTOR1,'IBTU    ',0,MAXBT2,0,0)     XECU0760
      IF(IRUXYZ.LE.0) CALL SETDS(IRUXYZ,ISTOR1,'RTUXYZ  ',1,8,MAXRTU,0) XECU0770
      IF(IBUXYZ.LE.0) CALL SETDS(IBUXYZ,ISTOR1,'BTUXYZ  ',1,8,MAXBTU,0) XECU0780
      IF(INVRTU.LE.0) CALL SETDS(INVRTU,ISTOR1,'INVRTU  ',0,MAXRTU,0,0) XECU0790
      IF(INVBTU.LE.0) CALL SETDS(INVBTU,ISTOR1,'INVBTU  ',0,MAXBTU,0,0) XECU0800
      IF(IRSTAT.LE.0) CALL SETDS(IRSTAT,ISTOR1,'IRSTAT  ',0,JRU,0,0)    XECU0810
      IF(IBSTAT.LE.0) CALL SETDS(IBSTAT,ISTOR1,'IBSTAT  ',0,JBU,0,0)    XECU0820
      IF(LBRSYS.LE.0) CALL SETDS(LBRSYS,ISTOR1,'LBRSYS  ',0,LENSYS,0,0) XECU0830
      IF(INFORM.LE.0) CALL SETDS(INFORM,ISTOR1,'INFORM  ',0,LENIFO,0,0) XECU0840
      KBUSG=ISTOR1(KBUSEG)+1                                            XECU0850
      KRUSG=ISTOR1(KRUSEG)+1                                            XECU0860
      ITBSG=ISTOR1(ITBSEG)+1                                            XECU0870
      ITRSG=ISTOR1(ITRSEG)+1                                            XECU0880
      INFOR=ISTOR1(INFORM)+1                                            XECU0890
      KBRSYS=ISTOR1(LBRSYS)+1                                           XECU0900
      LBSYS=ISTOR1(LBSYST)+1                                            XECU0910
      LRSYS=ISTOR1(LRSYST)+1                                            XECU0920
      IRT1=ISTOR1(IRTU)+1                                               XECU0930
      IBT1=ISTOR1(IBTU)+1                                               XECU0940
      IRUXY1=MOD(ISTOR1(IRUXYZ),10000)+1                                XECU0950
      IBUXY1=MOD(ISTOR1(IBUXYZ),10000)+1                                XECU0960
      INVRT1=ISTOR1(INVRTU)+1                                           XECU0970
      INVBT1=ISTOR1(INVBTU)+1                                           XECU0980
      IRSTA1=ISTOR1(IRSTAT)+1                                           XECU0990
      IBSTA1=ISTOR1(IBSTAT)+1
      CALL  ENCAGE(ISTOR1(KBUSG),ISTOR1(KRUSG),ISTOR1(ITBSG),
     1      ISTOR1(ITRSG),ISTOR1(INFOR),ISTOR1(KBRSYS),ISTOR1(LBSYS),
     2      ISTOR1(LRSYS),ISTOR1(IRT1),ISTOR1(IBT1),STORE1(IRUXY1),
     3      STORE1(IBUXY1),ISTOR1(INVRT1),ISTOR1(INVBT1),
     4      ISTOR1(IRSTA1),ISTOR1(IBSTA1))
      LEFTDS=ISTOR1(5)-ISTOR1(4)
      LESS=LEFTDS
      ITU=KRU+KBU
      LSYSPU=(LENSYS-IBRS)/ITU
      NSYSPU=NSYSPU-LSYSPU
      LESS=LESS+LSYSPU*ITU
      LSEGPU=(MAXSEG-LSTSEG)/LENSEG/ITU
      NSEGPU=NSEGPU-LSEGPU
      LESS=LESS+LSEGPU*ITU*LENSEG
      IF(NBTU.EQ.0) NBTU=1
      IF(NRTU.EQ.0) NRTU=1
      LESS=LESS+13*(MAXBTU-NBTU+MAXRTU-NRTU)
      MAXBTU=NBTU
      MAXRTU=NRTU
      LDIMEN=LDIMEN-LESS
      KSAVE=LESS*4/1000
      CALL PAGE
      WRITE(N6,DIMENS)
      RETURN
      END


      SUBROUTINE  SETDS(LPOINT,ISTORE,NAME,ITYPE,IDIM,JDIM,KDIM)
      COMMON /STRCON/ JPK,IPK
      DIMENSION ISTORE(1),NAME(2)
      DIMENSION LINE(10), XLINE(10)
      EQUIVALENCE (LINE(1),XLINE(1))
      DATA N6/6/
      IF(KDIM.GT.210) GO TO 9
      IF(IDIM.GT.9999) GO TO 9
      IF(JDIM.GT.999) GO TO 9
      IF( ITYPE.NE.1 .AND. ITYPE.NE.0) GO TO 9
      ISTOP=0
      LDIMEN=ISTORE(3)
      LASTA=ISTORE(4)
      LAST=ISTORE(5)
      NPOINT=ISTORE(6)
      LHEAD=MOD(ISTORE(7),IPK)
    3 LZERO=LASTA+LHEAD
      IF(KDIM.LE.0) GO TO 30
      IF(IDIM.GT.999 .OR. JDIM.GT.210) GO TO 9
      LENGTH=KDIM*JDIM*IDIM
      IPOINT=LZERO+IDIM*IPK+JDIM*JPK
      GO TO 4
   30 IF(JDIM.LE.0) GO TO 33
      LENGTH=JDIM*IDIM
      IPOINT=LZERO+IDIM*IPK
      GO TO 4
   33 LENGTH=IDIM                                                       SETD0270
      IPOINT=LZERO                                                      SETD0280
    4 NEED=LZERO+LENGTH                                                 SETD0290
      IF(NEED.GE.LAST) GO TO 2                                          SETD0300
      K=LDIMEN+1                                                        SETD0310
      IF(NPOINT.LE.0) GO TO 401                                         SETD0320
      DO 400 L=1,NPOINT                                                 SETD0330
      LPOINT=K-L                                                        SETD0340
      IF(ISTORE(LPOINT).EQ.0) GO TO 410                                 SETD0350
  400 CONTINUE                                                          SETD0360
  401 NPOINT=NPOINT+1                                                   SETD0370
      LPOINT=LAST                                                       SETD0380
      LAST=LAST-1                                                       SETD0390
  410 ISTORE(LPOINT)=IPOINT                                             SETD0400
      ISTORE(LZERO-4)=ITYPE+LPOINT*IPK                                  SETD0410
      ISTORE(LZERO-3)=NAME(1)                                           SETD0420
      ISTORE(LZERO-2)=NAME(2)                                           SETD0430
      ISTORE(LZERO-1)=0                                                 SETD0440
      ISTORE(LZERO)=IDIM+JDIM*IPK+KDIM*JPK                              SETD0450
      ISTORE(4)=NEED                                                    SETD0460
      ISTORE(5)=LAST                                                    SETD0470
      ISTORE(6)=NPOINT                                                  SETD0480
      RETURN                                                            SETD0490
    2 LABEL=ISTORE(7)/IPK                                               SETD0500
      MOVE=0                                                            SETD0510
      LASTI=LABEL                                                       SETD0520
      LASTJ=LABEL                                                       SETD0530
   20 JZERO=LASTJ+LHEAD                                                 SETD0540
      IF(LASTJ.GE.LASTA) GO TO 200                                      SETD0550
      IZERO=LASTI+LHEAD                                                 SETD0560
      ID=ISTORE(JZERO)                                                  SETD0570
      IF(ID.LE.0) GO TO 200                                             SETD0580
      IDI =MOD(ID,IPK)                                                  SETD0590
      JDI =MOD(ID,JPK)/IPK                                              SETD0600
      IF(JDI .LE.0) JDI =1                                              SETD0610
      KDI =ID/JPK                                                       SETD0620
      IF(KDI .LE.0) KDI =1                                              SETD0630
      LENG=IDI*JDI*KDI                                                  SETD0640
      LOAD=LENG  +LHEAD                                                 SETD0650
      MARK=ISTORE(JZERO-4)/IPK                                          SETD0660
      IF(MARK.GT.0) GO TO 230                                           SETD0670
      MOVE=MOVE+1                                                       SETD0680
      LASTJ=LASTJ+LOAD                                                  SETD0690
      GO TO 20                                                          SETD0700
  230 IF(MOVE.GT.0) GO TO 240                                           SETD0710
      LASTJ=LASTJ+LOAD                                                  SETD0720
      LASTI=LASTI+LOAD                                                  SETD0730
      GO TO 20                                                          SETD0740
  240 DO 250 L=1,LOAD                                                   SETD0750
      LASTJ=LASTJ+1                                                     SETD0760
      LASTI=LASTI+1                                                     SETD0770
  250 ISTORE(LASTI)=ISTORE(LASTJ)                                       SETD0780
      ID=ISTORE(MARK)/IPK                                               SETD0790
      ISTORE(MARK)=ID*IPK+IZERO                                         SETD0800
      GO TO 20                                                          SETD0810
  200 LASTJ=LASTI+1
      IF(LASTJ.GT.LASTA) GO TO 220
      DO 210 L=LASTJ,LASTA
  210 ISTORE(L)=0
  220 ISTORE(4)=LASTI
      LASTA=LASTI
      LEFT=LAST-LASTA
      LREQD=LENGTH+LHEAD+1
      IF(LEFT.GE.LREQD) GO TO 3
      ISTOP=1
      WRITE(N6,6010) ISTORE(1),ISTORE(2),LEFT,LREQD,NAME
      ENTRY     DUMPDS(ISTORE)
      NPOINT=ISTORE(6)
      IF(NPOINT.LE.0) GO TO 112
      IF (NPOINT.GT.500) GO TO 112
      LABEL=ISTORE(7)/IPK
      WRITE(N6,6100) (ISTORE(L),L=1,LABEL)
      M=ISTORE(3)+1
      N=1
   10 LP=M-N
      IPOINT=ISTORE(LP)
      IF(IPOINT) 11,11,111
      ENTRY     WRITDS(LPOINT,ISTORE)
      NPOINT=1
      N=1
      IF(LPOINT.LE.0) GO TO 11
      IPOINT=ISTORE(LPOINT)
  111 IZO=MOD(IPOINT,IPK)
      IF(IZO.LE.0) GO TO 11
      N1    =ISTORE(IZO-3)
      N2    =ISTORE(IZO-2)
      LDIM=ISTORE(IZO)
      IDI =MOD(LDIM,IPK)
      JDI =MOD(LDIM,JPK)/IPK
      ITYPE=MOD(ISTORE(IZO-4),10)
      IFILL=ISTORE(IZO-1)
      IF(IFILL.EQ.0) IFILL=LDIM
      KPR=IFILL/JPK
      IF(KPR.EQ.0) KPR=1
      JPR=MOD(IFILL,JPK)/IPK
      IF(JPR.EQ.0) JPR=1
      IPR=MOD(IFILL,IPK)
      IF(IPR.EQ.0) GO TO 11
      ASSIGN 121 TO LN
      L=0
      WRITE(N6,6600)N1,N2,IPOINT,LPOINT,ISTORE(1),ISTORE(2)
      DO 100 K=1,KPR
      KADD=(K-1)*JDI*IDI+IZO
      DO 110 J=1,JPR
      JKADD=(J-1)*IDI+KADD
      DO 120 I=1,IPR
      LOC=JKADD+I
      GO TO LN,(121,125)
  121 KP=K
      JP=J
      IP=I                                                              SETD1370
      ASSIGN 125 TO LN                                                  SETD1380
  125 L=L+1                                                             SETD1390
      LINE(L)=ISTORE(LOC)                                               SETD1400
      IF(L.LT.10) GO TO 120                                             SETD1410
      IF(ITYPE) 1000,1000,1111                                          SETD1420
 1000 WRITE(N6,6111) IP,JP,KP,LINE                                      SETD1430
      GO TO 130                                                         SETD1440
 1111 WRITE(N6,6111) IP,JP,KP,XLINE                                     SETD1450
  130 L=0                                                               SETD1460
      ASSIGN 121 TO LN                                                  SETD1470
  120 CONTINUE                                                          SETD1480
  110 CONTINUE                                                          SETD1490
  100 CONTINUE                                                          SETD1500
      IF(L.EQ.0) GO TO 150                                              SETD1510
      IF(ITYPE) 1001,1001,1110                                          SETD1520
 1001 WRITE(N6,6111) IP,JP,KP,(LINE(I),I=1,L)                           SETD1530
      GO TO 150                                                         SETD1540
 1110 WRITE(N6,6111) IP,JP,KP,(XLINE(I),I=1,L)                          SETD1550
  150 WRITE(N6,6601) IPR,JPR,KPR                                        SETD1560
   11 N=N+1                                                             SETD1570
      IF(N.LE.NPOINT) GO TO 10                                          SETD1580
  112 IF(ISTOP.GT.0) STOP                                               SETD1590
      RETURN                                                            SETD1600
    9 WRITE(N6,6000) ITYPE,IDIM,JDIM,KDIM,NAME,ISTORE(1),ISTORE(2)      SETD1610
      STOP                                                              SETD1620
      ENTRY      FREEDS(LPOINT,ISTORE)                                  SETD1630
      IF(LPOINT.LE.0) RETURN                                            SETD1640
      LZERO=MOD(ISTORE(LPOINT),IPK)                                     SETD1650
      K=ISTORE(LZERO-4)                                                 SETD1660
      IF(K.GT.0) K=-K                                                   SETD1670
      ISTORE(LZERO-4)=K                                                 SETD1680
      ISTORE(LPOINT)=0                                                  SETD1690
      LPOINT=0                                                          SETD1700
      RETURN                                                            SETD1710
 6000 FORMAT(1H1//' STORAGE ARRAY CREATION VARIABLES INCOMPATIBLE'/     SETD1720
     1     '  ITYPE = ',I5/'  IDIM = ',I5/'  JDIM = ',I5/'  KDIM = ',   SETD1730
     2     '  CREATED ARRAY NAME = ',2A4/'  STORAGE ARRAY = ',2A4  )    SETD1740
 6010 FORMAT(1H1//' STORAGE ARRAY = ',2A4, ' HAS ',I5,' WORDS LEFT',    SETD1750
     1     ' AND IS EXCEEDED BY THE ',I5,' WORDS REQUIRED ',            SETD1760
     2     'BY THE CREATED ARRAY = ',2A4)                               SETD1770
 6100 FORMAT(1H1/' DUMP OF STORAGE ARRAY = ',2A4/' LABEL=',(12I10))     SETD1780
 6111 FORMAT(1X,I4,2I3,')',10G12.5)                                     SETD1790
 6600 FORMAT(//'   I   J   K )  VALUES FOR ARRAY = ',2A4,5X,'MAP POINT',SETD1800
     1     'ER = ',I10,'  AT LOC',I5,' IN STORAGE ARRAY = ',2A4)        SETD1810
 6601 FORMAT( '   PRINT DIMENSIONS = ', 3I5)                            SETD1820
      END                                                               SETD1830


      SUBROUTINE ENGAGE(KBUSEG,KRUSEG,ITBSEG,ITRSEG,INFORM,LBRSYS,      ENGA0010
     1     LBSYST,LRSYST,IRTU,IBTU,RTUXYZ,BTUXYZ,INVRTU,INVBTU,         ENGA0020
     2     IRSTAT,IBSTAT)                                               ENGA0030
      DIMENSION KBUSEG(1),KRUSEG(1),ITBSEG(1),ITRSEG(1),INFORM(1),      ENGA0040
     1     LBRSYS(1),LBSYST(1),LRSYST(1)                                ENGA0050
```

```
      DIMENSION IRTU(1),IBTU(1),RTUXYZ(8,1),BTUXYZ(8,1),INVRTU(1),      ENGA0060
     1     INVBTU(1),IRSTAT(1),IBSTAT(1)                                 ENGA0070
      COMMON/BATTRY/ NBATT,MISRAD(2,10)                                  ENGA0080
CKILLF                                                                   ENGA0090
       COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,         ENGA0100
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULCST(100),        ENGA0110
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                         ENGA0120
CNAVIG                                                                   ENGA0130
       COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                  ENGA0140
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),       ENGA0150
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),       ENGA0160
     3          TTIME,NUMX,                                              ENGA0170
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),    ENGA0180
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)     ENGA0190
CPLAT   VS  SYSTEMS  (SUBSYSTEMS)                                        ENGA0200
       COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),           ENGA0210
     1   NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),         ENGA0220
     2   BPX1(12,15),                                                    ENGA0230
     3                 NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),        ENGA0240
     4   NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),         ENGA0250
     5   RPX1(12,15)                                                     ENGA0260
CMSLSYS                          16    2     2     6     6    24    50 ENGA0270
       COMMON/CMSLSY/ NMSL,NMSLMX,NBSSM,NBASM,NRSSM,NRASM,LATMSL,LTRAJ,  ENGA0280
     1    NAMMSL(2,16),KMSLTY(16),                                       ENGA0290
     2    DATMSL(24,16), TTRAJ(50,16)                                    ENGA0300
CZZSYS                             25     11     12   15                ENGA0310
       COMMON/CZZSYS/NZZSYS,NZZSMX,NBZSYS,NRZSYS,LAZZ,NAMSYS(2,25),      ENGA0320
     1    KZZTYP(25),ZZSYS(15,25)                                        ENGA0330
CSAMLT               8,12,15,                                            ENGA0340
      COMMON/CSAMLT/LX,MX,NX,RMN(15),RMX(15),EMN(15),EMX(15),ZMX(15),    ENGA0350
     1 XMX(15),  RT(8,12,15)                                             ENGA0360
CSASYS                               15    4     4     2     4    20     ENGA0370
       COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,       ENGA0380
     1    NAMSA(2,15),KSATYP(15),SASYS(20,15)                            ENGA0390
CRADAR                                                                   ENGA0400
       COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                          ENGA0410
     1   BRAD(36,17),BENV(6,2),BFTC(4,2),BTAR(6,3),BJAM(6,3,2),          ENGA0420
     2                NRSR,NRTR,NRRMX,NRJ,NRJMX,                         ENGA0430
     3   RRAD(36,15),RENV(6,2),RFTC(4,2),RTAR(6,3),RJAM(6,3,2)           ENGA0440
       COMMON/CHOMER/ NHOM,NHOMMX,NAMHOM(2,10),KHOMTY(10),DATHOM(36,10) ENGA0450
CSONAR                           15     5     8     12                  ENGA0460
       COMMON/CSONAR/NSON,NSONMX,NBSON,NRSON,LASON,NAMSON( 2,15),        ENGA0470
     1    KSONTY(15),  SONSYS(12,15)                                     ENGA0480
      COMMON /SONCRV/ IBA,IBB,IDA,IDB,IBOTTM,IACTIV,SRANGE(17),          ENGA0490
     1     SMERIT(17)                                                    ENGA0500
      COMMON/STRCON/  JPK,IPK                                            ENGA0510
      COMMON/INFO/  LENIFO,LENSEG,MAXSEG,NEXSEG,LSTSEG,LZCSEG            ENGA0520
      COMMON/GNAV/ BGCV(8,5,18),RGCV(8,5,18), INTVLB(38),INTVLR(38)      ENGA0530
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              ENGA0540
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP          ENGA0550
      COMMON/ECONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,EDMIN,           ENGA0560
     1     AIRCPT,EPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)          ENGA0570
      COMMON/EXSTOR/ LENSYS,MAXRTU,MAXBTU,LEFTDS,NRTU,NBTU,IBRS,NPHASE   ENGA0580
      COMMON/CTLOOP/ IDU1,IPRINT,JPRINT,IVEC,LTIMES,IBVEC,IRVEC,         ENGA0590
     1     IBTVEC,IRTVEC,LEFTB,LEFTR,NEXBTU,NEXRTU,PRSTEP,PSTART,        ENGA0600
     2      PSTOP,SSTEP,TPRINT,TLAST,TIMES(100)                            ENGA0610
C                                                                           ENGA0620
      DIMENSION JWP(4),WEP(5),JTG(5),TGT(5)                                 ENGA0630
      EQUIVALENCE (IMISC(3),NLPR), (IMISC(4),ISIDE), (IMISC(5),KILLPR)      ENGA0640
     1,     (IMISC(7),ISNAPR), (IMISC(11),IVECB), (IMISC(12),IVECR)         ENGA0650
     2,     (IMISC(13),IRNDVB), (IMISC(14),IRNDVR), (IMISC(29),IRSAM)       ENGA0660
     3,     (IMISC(31),NXSHIP), (IMISC(32),NXAC), (IMISC(33),NXSUB)         ENGA0670
     4,     (IMISC(34),NXCM)                                                ENGA0680
      DATA JWP,JTG,WEP,TGT/9*0,         10*0./                              ENGA0690
C                                                                           ENGA0700
      NAMELIST /NGAGE/ NCARLO,TBEGIN,TIMEND,TSTEP,NRANDM,NPHASE,            ENGA0710
     1     IPRINT,ISNAPR,PRSTEP,PSTART,PSTOP,NLPR,KILLPR,                   ENGA0720
     2     PDCLAS,HVTB,HVTR,HVTIME,IVECB,IVECR,IRNDVB,IRNDVR,IRSAM,         ENGA0730
     3     NXSHIP,NXAC,NXSUB,NXCM,                                          ENGA0740
     4     IPRAD,IENV,JAM,ISCAN,PDTIME,PDMIN,PDTMIN,PDCMIN,IACTIV,          ENGA0750
     5     IBOTTM,AIRCPT,IMISC,NBATT,MISRAD,NTIMES,TIMES                    ENGA0760
C                                                                           ENGA0770
 6401 FORMAT(//23X,' CRUISE     HITS ON     MISSILES',                      ENGA0780
     1      /23X,' MISSILES  TARGETS    KILLED',                            ENGA0790
     2      /23X,' LAUNCHED  EVALUATED  ENROUTE' /)                         ENGA0800
 6402 FORMAT(/6X,2A4,I9,I6,I10,I10)                                         ENGA0810
C                                                                           ENGA0820
      DO 10 I=1,100                                                         ENGA0830
   10 TIMES(I)=0.                                                           ENGA0840
      ICU1=50                                                               ENGA0850
      TSTEP=.001                                                            ENGA0860
      PDTIME=.0201                                                          ENGA0870
      PRSTEP=.1                                                             ENGA0880
      JAM=0                                                                 ENGA0890
      NPHASE=6                                                              ENGA0900
      PDCMIN=.5                                                             ENGA0910
      NRANDM=1                                                              ENGA0920
      NCARLO=1                                                              ENGA0930
      IMISC(31)=5                                                           ENGA0940
      IMISC(32)=2                                                           ENGA0950
      IMISC(33)=2                                                           ENGA0960
      IMISC(34)=1                                                           ENGA0970
      HVTB=0.                                                               ENGA0980
      HVTR=0.                                                               ENGA0990
      PDCLAS=0.                                                             ENGA1000
      HVTIME=0.                                                             ENGA1010
      IPRINT=1                                                              ENGA1020
      TBEGIN=100.                                                           ENGA1030
      TIMEND=100.                                                           ENGA1040
      PSTART=-100.                                                          ENGA1050
      PSTOP=100.                                                            ENGA1060
      CALL PAGE                                                             ENGA1070
C                                                                           ENGA1080
C----   INPUT ENGAGEMENT CONTROL PARAMETERS                                 ENGA1090
C                                                                           ENGA1100
      READ(N5,NGAGE)                                                        ENGA1110
      IMISC(1)=PDTIME*1000.+.01                                             ENGA1120
      IMISC(2)=JAM                                                          ENGA1130
      IMISC(6)=PDCMIN*100.+.01                                              ENGA1140
      IMISC(9)=HVTB+.01                                                     ENGA1150
      IMISC(10)=HVTR+.01                                                ENGA1160
      IMISC(21)=PDCLAS*100.+.01                                         ENGA1170
      IMISC(28)=HVTIME*1000.+.01                                        ENGA1180
C                                                                       ENGA1190
C                                                                       ENGA1200
      CALL SONARM                                                       ENGA1210
      X=IENV                                                            ENGA1220
      BENV(1,1)=X+.01                                                   ENGA1230
      RENV(1,1)=X+.01                                                   ENGA1240
      SSTEP=TSTEP                                                       ENGA1250
      DO 100 I=1,NRANDM                                                 ENGA1260
  100 XRN=URN(IDUM)                                                     ENGA1270
      DO 500 ICARLO=1,NCARLO                                            ENGA1280
      INIT=1                                                            ENGA1290
      DO 110 I=1,16                                                     ENGA1300
      DO 110 J=21,24                                                    ENGA1310
  110 DATMSL(J,I)=0.                                                    ENGA1320
      CALL KILLEX(INIT,ISIDE,JWP,WEP,JTG,TGT,IPR,IPR)                   ENGA1330
      CALL GCDATA(BGC,BGCV,NBG,INTVLB,BE,BF)                            ENGA1340
      CALL GCDATA(RGC,RGCV,NRG,INTVLR,RE,RF)                            ENGA1350
      NEXSEG=0                                                          ENGA1360
      LSTSEG=1-LENSEG                                                   ENGA1370
      IBRS=0                                                            ENGA1380
      CALL SYSTEM(KBPXS,KBPT,KBST,NBP,62,NBSS,BRAD,LBRSYS,              ENGA1390
     1       IBRS,LENSYS,KBU,KBUK,LBSYST,KBGK,NBU,NBG,IBSTAT,           ENGA1400
     2       BPX1,SASYS,ZZSYS,DATMSL,AIRCPT,NVALT,IACTIV,SCNSYS,TTRAJ)  ENGA1410
      CALL SYSTEM(KRPXS,KRPT,KRST,NRP,82,NRSS,RRAD,LBRSYS,              ENGA1420
     1       IBRS,LENSYS,KRU,KRUK,LRSYST,KRGK,NRU,NRG,IRSTAT,           ENGA1430
     2       RPX1,SASYS,ZZSYS,DATMSL,AIRCPT,NVALT,IACTIV,SCNSYS,TTRAJ)  ENGA1440
      NBTU=0                                                            ENGA1450
      NRTU=0                                                            ENGA1460
      NEXBTU=0                                                          ENGA1470
      NEXRTU=0                                                          ENGA1480
      IF(ICARLO.GT.1) GO TO 30                                          ENGA1490
      MAXTIM=100                                                        ENGA1500
      TCLOSE=.001                                                       ENGA1510
      NTG=NBG+NRG                                                       ENGA1520
      NTIMES=1                                                          ENGA1530
      NOW=1                                                             ENGA1540
      TIMES(NTIMES)=BGC(2,1,1)                                          ENGA1550
      DO 840 NT=2,8                                                     ENGA1560
      DO 850 N=1,NTG                                                    ENGA1570
      I=N-NBG                                                           ENGA1580
      IF(I) 81,81,810                                                   ENGA1590
   81 TRIAL=BGC(NT,1,N)                                                 ENGA1600
      GO TO 811                                                         ENGA1610
  810 TRIAL=RGC(NT,1,I)                                                 ENGA1620
  811 LAST=0                                                            ENGA1630
  815 TNOW=TIMES(NOW)                                                   ENGA1640
      DIFF=TRIAL-TNOW                                                   ENGA1650
      IF(DIFF) 80,850,82                                                ENGA1660
   80 IF(ABS(DIFF).LT.TCLOSE) GO TO 850                                 ENGA1670
      IF(LAST) 802,801,83                                               ENGA1680
  801 LAST=-1                                                           ENGA1690
  802 IF(NOW.EQ.1) GO TO 83                                             ENGA1700
```

```
      NOW=NOW-1                                                         ENGA1710
      GO TO 815                                                         ENGA1720
   82 IF(ABS(DIFF).LT.TCLOSE) GO TO 850                                 ENGA1730
      NOW=NOW+1                                                         ENGA1740
      IF(LAST) 83,821,822                                               ENGA1750
  821 LAST=1                                                            ENGA1760
  822 IF(NOW-NTIMES) 815,815,831                                        ENGA1770
   83 K=NTIMES+1                                                        ENGA1780
      DO 830 I=NOW,NTIMES                                               ENGA1790
      TIMES(K)=TIMES(K-1)                                               ENGA1800
  830 K=K-1                                                             ENGA1810
  831 NTIMES=NTIMES+1                                                   ENGA1820
      TIMES(NOW)=TRIAL                                                  ENGA1830
      IF(NTIMES.LE.MAXTIM) GO TO 850                                    ENGA1840
      CALL PAGE                                                         ENGA1850
      WRITE(N6,6801) MAXTIM,TIMES                                       ENGA1860
 6801 FORMAT(//6X,'NUMBER OF TIMES EXCEEDS',I5/(6X,10G12.5))            ENGA1870
      STOP                                                              ENGA1880
  850 CONTINUE                                                          ENGA1890
  840 CONTINUE                                                          ENGA1900
      IF(TBEGIN.EQ.100.) TBEGIN=TIMES(1)                                ENGA1910
      TIMAX=TBEGIN+21.000                                               ENGA1920
      IF(TIMEND.EQ.100.) TIMEND=TIMES(NTIMES)                           ENGA1930
      IF(TIMEND.GT.TIMAX) TIMEND=TIMAX                                  ENGA1940
      LTIMES=NTIMES-1                                                   ENGA1950
      WRITE(N6,NGAGE)                                                   ENGA1960
   30 TLAST=TIMES(1)-SSTEP                                              ENGA1970
      TPRINT=TBEGIN                                                     ENGA1980
C                                                                       ENGA1990
      CALL TILOOP(KBUSEG,KRUSEG,ITBSEG,ITRSEG,INFORM,LBRSYS,             ENGA2000
     1     LBSYST,LRSYST,IRTU,IBTU,RTUXYZ,BTUXYZ,INVRTU,INVBTU,         ENGA2010
     2     IRSTAT,IBSTAT)                                               ENGA2020
C                                                                       ENGA2030
  400 CONTINUE                                                          ENGA2040
      IF(NPHASE.LT.2) RETURN                                            ENGA2050
      IVEC=54321                                                        ENGA2060
      CALL ALLXYZ(TIME,      1,N6,IVEC)                                 ENGA2070
      CALL SUMOUT(ICARLO)                                               ENGA2080
      CALL PAGE                                                         ENGA2090
      WRITE(N6,6401)                                                    ENGA2100
      DO 401 I=1,16                                                     ENGA2110
      L=DATMSL(22,I)+.5                                                 ENGA2120
      IF (L.EQ.0) GO TO 401                                             ENGA2130
      J=DATMSL(23,I)+.5                                                 ENGA2140
      K=DATMSL(24,I)+.5                                                 ENGA2150
      WRITE(N6,6402) NAMMSL(1,I),NAMMSL(2,I),KMSLTY(I),L,J,K            ENGA2160
  401 CONTINUE                                                          ENGA2170
      DO 402 I=1,KBU                                                    ENGA2180
      KBUK(4,I)=1                                                       ENGA2190
  402 KBUSEG(I)=0                                                       ENGA2200
      DO 403 I=1,KRU                                                    ENGA2210
      KRUK(4,I)=1                                                       ENGA2220
  403 KRUSEG(I)=0                                                       ENGA2230
      K=KBU+MAXBTU                                                      ENGA2240
      DO 404 I=1,K                                                      ENGA2250
 404  ITBSEG(I)=0                                                   ENGA2260
      K=KRU+MAXRTU                                                  ENGA2270
      DO 405 I=1,K                                                  ENGA2280
 405  ITRSEG(I)=0                                                   ENGA2290
      K=2*KBU+11                                                    ENGA2300
      DO 406 I=1,K                                                  ENGA2310
 406  LBSYST(I)=0                                                   ENGA2320
      K=2*KRU+11                                                    ENGA2330
      DO 407 I=1,K                                                  ENGA2340
 407  LRSYST(I)=0                                                   ENGA2350
 500  CONTINUE                                                      ENGA2360
      RETURN                                                        ENGA2370
      END                                                           ENGA2380


      SUBROUTINE TILOOP(KBUSEG,KRUSEG,ITBSEG,ITRSEG,INFORM,LBRSYS,      TIL00010
     1      LBSYST,LRSYST,IRTU,IBTU,RTUXYZ,BTUXYZ,INVRTU,INVBTU,        TIL00020
     2      IRSTAT,IBSTAT)                                              TIL00030
      DIMENSION KBUSEG(1),KRUSEG(1),ITBSEG(1),ITRSEG(1),INFORM(1),      TIL00040
     1      LBRSYS(1),LBSYST(1),LRSYST(1)                               TIL00050
      DIMENSION IRTU(1),IBTU(1),RTUXYZ(8,1),BTUXYZ(8,1),INVRTU(1),      TIL00060
     1      INVBTU(1),IRSTAT(1),IBSTAT(1)                               TIL00070
      COMMON/BATTRY/ NBATT,MISRAD(2,10)                                 TIL00080
CKILLF                                                                  TIL00090
       COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,        TIL00100
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULCST(100),       TIL00110
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                        TIL00120
CNAVIG                                                                  TIL00130
       COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                 TIL00140
     1      NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),     TIL00150
     2      NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),     TIL00160
     3          TTIME,NUMX,                                             TIL00170
     4      KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),  TIL00180
     5      KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)   TIL00190
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                     TIL00200
       COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),          TIL00210
     1   NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),        TIL00220
     2   BPX1(12,15),                                                   TIL00230
     3               NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),         TIL00240
     4   NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),        TIL00250
     5   RPX1(12,15)                                                    TIL00260
CMSLSYS                        16     2     2     6     6     24     50 TIL00270
       COMMON/CMSLSY/ NMSL,NMSLMX,NBSSM,NBASM,NRSSM,NRASM,LATMSL,LTRAJ, TIL00280
     1    NAMMSL(2,16),KMSLTY(16),                                      TIL00290
     2    DATMSL(24,16), TTRAJ(50,16)                                   TIL00300
CZZSYS                         25     11     12    15                   TIL00310
       COMMON/CZZSYS/NZZSYS,NZZSMX,NBZSYS,NRZSYS,LAZZ,NAMSYS(2,25),     TIL00320
     1    KZZTYP(25),ZZSYS(15,25)                                       TIL00330
CSAMLT                  8,12,15,                                        TIL00340
      COMMON/CSAMLT/LX,MX,NX,RMN(15),RMX(15),EMN(15),EMX(15),ZMX(15),   TIL00350
     1 XMX(15),   RT(8,12,15)                                           TIL00360
CSASYS                         15     4     4      2     4     20       TIL00370
       COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,      TIL00380
     1    NAMSA(2,15),KSATYP(15),SASYS(20,15)                           TIL00390
```

```
CRADAR                                                                  TIL00400
      COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                          TIL00410
     1   BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,3,2),         TIL00420
     2                NRSR,NRTR,NRRMX,NRJ,NRJMX,                        TIL00430
     3   RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,3,2)          TIL00440
      COMMON/CHOMER/ NHOM,NHOMMX,NAMHOM(2,10),KHOMTY(10),DATHOM(36,10)  TIL00450
CSONAR                            15     5     8     12                 TIL00460
      COMMON/CSONAR/NSON,NSONMX,NBSCN,NRSCN,LASON,NAMSCN( 2,15),        TIL00470
     1    KSONTY(15),  SONSYS(12,15)                                    TIL00480
      COMMON /SONCRV/ IPA,IBB,IDA,IDB,IBOTTM,IACTIV,SRANGE(17),         TIL00490
     1      SMERIT(17)                                                  TIL00500
      COMMON/STRCON/   JPK,IPK                                          TIL00510
      COMMON/INFO/  LENIFO,LENSEG,MAXSEG,NEXSEG,LSTSEG,LZOSEG           TIL00520
      COMMON/GNAV/ BGCV(8,5,18),RGCV(8,5,18), INTVLB(38),INTVLR(38)     TIL00530
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             TIL00540
      COMMON /FTIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP         TIL00550
      COMMON/FCONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,FDMIN,          TIL00560
     1      AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)        TIL00570
      COMMON/EXSTOR/ LENSYS,MAXRTU,MAXBTU,LEFTDS,NRTU,NBTU,IBRS,NPHASE  TIL00580
      COMMON/CTLOOP/ IDU1,IPRINT,JPRINT,IVEC,LTIMES,IBVEC,IRVEC,        TIL00590
     1      IBTVEC,IRTVEC,LEFTB,LEFTR,NEXBTU,NEXRTU,PRSTEP,PSTART,      TIL00600
     2      PSTOP,SSTEP,TPRINT,TLAST,TIMES(100)                         TIL00610
      DIMENSION KRHVT(50),KBHVT(50)                                     TIL00620
C                                                                       TIL00630
      EQUIVALENCE (ISIDE,IMISC(4)),(ISNAPR,IMISC(7)),                   TIL00640
     1      (NLPR,IMISC(3)),(KILLPR,IMISC(5))                           TIL00650
C                                                                       TIL00660
      NAMELIST /INTLEG/  INTVLB,INTVLR,KBHVT,KRHVT                      TIL00670
C                                                                       TIL00680
      IBNV=1                                                            TIL00690
      IRNV=1                                                            TIL00700
      IF(IMISC(28).EQ.0) IMISC(28)=22000                                TIL00710
      DO 110 I=1,50                                                     TIL00720
      KRHVT(I)=0                                                        TIL00730
  110 KBHVT(I)=0                                                        TIL00740
      DO 395 KTIME=1,LTIMES                                             TIL00750
      TIMEA=TIMES(KTIME)                                                TIL00760
      TIMEB=TIMES(KTIME+1)                                              TIL00770
      TIME=TIMEA                                                        TIL00780
      IEVTIM=(TIME-TBEGIN)*1000.                                        TIL00790
      JPRINT=IPRINT                                                     TIL00800
    3 IF(TIME.LT.TBEGIN) GO TO 390                                      TIL00810
      IF(TIME.GT.TIMEND) RETURN                                         TIL00820
      IF(TIME.GE.TPRINT) JPRINT=IPRINT                                  TIL00830
      ITIME=(TIME-TBEGIN)*1000.                                         TIL00840
      TSTEP=TIME-TLAST                                                  TIL00850
      IVEC=54321                                                        TIL00860
      CALL ALLXYZ(TIME,JPRINT,N6,IVEC)                                  TIL00870
      IF(JPRINT.EQ.0) GO TO 300                                         TIL00880
      IF(IMISC(3).GT.0) WRITE(N6,INTLEG)                                TIL00890
  300 IBVEC=IVEC/10                                                     TIL00900
      ISIDE=1                                                           TIL00910
      IRVEC=MOD(IVEC,10)                                                TIL00920
      IRTVEC=0                                                          TIL00930
      LEFTB=0                                                           TIL00940
      IF(NBTU.EQ.0) GO TO 305                                           TIL00950
      CALL  TUXYZ(NBTU,IBTU,INVBTU,BTUXYZ,MAXBTU,                       TIL00960
     A      KBUK,INTVLB,BXYZ,BGCV,                                      TIL00970
     1      KRU,INTVLR,KRUK,RXYZ,RGCV,                                  TIL00980
     2      TTRAJ,IVEC,LEFTB,IDU1)                                      TIL00990
      IF(IVEC.GT.0) IBTVEC=IVEC                                         TIL01000
  305 IRTVEC=0                                                          TIL01010
      ISIDE=2                                                           TIL01020
      LEFTR=0                                                           TIL01030
      IF(NRTU.EQ.0) GO TO 310                                           TIL01040
      CALL  TUXYZ(NRTU,IRTU,INVRTU,RTUXYZ,MAXRTU,                       TIL01050
     A      KRUK,INTVLR,RXYZ,RGCV,                                      TIL01060
     1      KBU,INTVLB,KBUK,BXYZ,BGCV,                                  TIL01070
     3      TTRAJ,IVEC,LEFTR,IDU1)                                      TIL01080
      IF(IVEC.GT.0) IRTVEC=IVEC                                         TIL01090
  310 IF(LEFTB.EQ.0) GO TO 315                                          TIL01100
      ISIDE=1                                                           TIL01110
      CALL  TU2XYZ(NBTU,IBTU,INVBTU,BTUXYZ,                             TIL01120
     1      KRU,INVRTU,RTUXYZ,                                          TIL01130
     2      IVEC,IDU1)                                                  TIL01140
      IF(IVEC.GT.0) IBTVEC=IVEC                                         TIL01150
  315 IF(LEFTR.EQ.0) GO TO 320                                          TIL01160
      ISIDE=2                                                           TIL01170
      CALL  TU2XYZ(NRTU,IRTU,INVRTU,RTUXYZ,                             TIL01180
     1      KBU,INVBTU,BTUXYZ,                                          TIL01190
     2      IVEC,IDU1)                                                  TIL01200
  320 IF(NPHASE.LT.3) GO TO 380                                         TIL01210
      ISIDE=1                                                           TIL01220
      IF(IBVEC+IRVEC+IRTVEC  .EQ. 0) GO TO 325                          TIL01230
      CALL  RELATE(NBG,NBU,INTVLB,LBSYST,KBUSEG,                        TIL01240
     1      BGCV,BXYZ,BPX1,IBSTAT,                                      TIL01250
     2      KRU,NRG,NRTU,INTVLR,INVRTU,NRU,LRSYST,ITRSEG,               TIL01260
     3      IRTU,RPX1,RGCV,KRUK,RXYZ,RTUXYZ,IRVEC,IRTVEC,IRSTAT,        TIL01270
     4      INFORM,LBRSYS,DATMSL,IDU1)                                  TIL01280
  325 IF(IRVEC+IBVEC+IBTVEC  .EQ. 0) GO TO 330                          TIL01290
      ISIDE=2                                                           TIL01300
      CALL  RELATE(NRG,NRU,INTVLR,LRSYST,KRUSEG,                        TIL01310
     1      RGCV,RXYZ,RPX1,IRSTAT,                                      TIL01320
     2      KBU,NBG,NBTU,INTVLB,INVBTU,NBU,LBSYST,ITBSEG,               TIL01330
     3      IBTU,BPX1,BGCV,KBUK,BXYZ,BTUXYZ,IBVEC,IBTVEC,IBSTAT,        TIL01340
     4      INFORM,LBRSYS,DATMSL,IDU1)                                  TIL01350
  330 IF(NPHASE.LT.4) GO TO 380                                         TIL01360
      ISIDE=1                                                           TIL01370
      CALL  DETECT(KBU,NBTU,LBSYST,KBUSEG,BRAD,BXYZ,KBUK,INTVLB,        TIL01380
     1      BGCV,BPX1,BENV(1,IBNV),MAXBTU,IBSTAT,BTUXYZ,KBST,IBTU,      TIL01390
     2      KRU,NRTU,LRSYST,ITRSEG,RXYZ,KRUK,INTVLR,RGCV,RPX1,          TIL01400
     3      RJAM,IRTU,RTUXYZ,IRSTAT,KRHVT,                              TIL01410
     4      DATMSL,LBRSYS,INFORM,DATHOM,        IDU1)                   TIL01420
      ISIDE=2                                                           TIL01430
      CALL  DETECT(KRU,NRTU,LRSYST,KRUSEG,RRAD,RXYZ,KRUK,INTVLR,        TIL01440
     1      RGCV,RPX1,RENV(1,IRNV),MAXRTU,IRSTAT,RTUXYZ,KRST,IRTU,      TIL01450
     2      KBU,NBTU,LBSYST,ITBSEG,BXYZ,KBUK,INTVLB,BGCV,BPX1,          TIL01460
     3      BJAM,IBTU,BTUXYZ,IBSTAT,KBHVT,                              TIL01470
     4      DATMSL,LBRSYS,INFORM,DATHOM,        IDU1)                   TIL01480
      IF(NPHASE.LT.5) GO TO 380                                         TIL01490
      IF(ITIME.EQ.0) GO TO 380                                          TIL01500
      IF(IHVTIM.GT.ITIME) GO TO 3333                                    TIL01510
      IHVTIM=IHVTIM+IMISC(28)                                           TIL01520
      CALL  HVTGT(IMISC(11),NBG,INTVLB,BGC,BGCV,                        TIL01530
     1      KRU,IRSTAT,KRUK,RXYZ,RREL,IMISC(10),KRHVT,                  TIL01540
     2      TIME    )                                                   TIL01550
      CALL  HVTGT(IMISC(12),NRG,INTVLR,RGC,RGCV,                        TIL01560
     1      KBU,IBSTAT,KBUK,BXYZ,BREL,IMISC(9), KBHVT,                  TIL01570
     2      TIME    )                                                   TIL01580
 3333 CONTINUE                                                          TIL01590
      ISIDE=1                                                           TIL01600
      CALL  ALLOCA(NEXBTU,NBTU,MAXBTU,IBSTAT,LBSYST,IBTU,INVBTU,        TIL01610
     A      BTUXYZ,BPX1,BXYZ,BRAD,BENV(1,IBNV),KBUSEG,KBU,              TIL01620
     1      KBUK,NBU,                                                   TIL01630
     2      KRU,NRTU,ITRSEG,IRSTAT,INTVLR,RTUXYZ,KRUK,RGCV,RXYZ,KRHVT,  TIL01640
     3      RJAM,IRTU,RPX1,LRSYST,INVRTU,                               TIL01650
     4      TTRAJ,DATMSL,ZZSYS,SASYS,LBRSYS,INFORM,IDU1)                TIL01660
      ISIDE=2                                                           TIL01670
      CALL  ALLOCA(NEXRTU,NRTU,MAXRTU,IRSTAT,LRSYST,IRTU,INVRTU,        TIL01680
     A      RTUXYZ,RPX1,RXYZ,RRAD,RENV(1,IRNV),KRUSEG,KRU,              TIL01690
     1      KRUK,NRU,                                                   TIL01700
     2      KBU,NBTU,ITBSEG,IBSTAT,INTVLB,BTUXYZ,KBUK,BGCV,BXYZ,KBHVT,  TIL01710
     3      BJAM,IBTU,BPX1,LBSYST,INVBTU,                               TIL01720
     4      TTRAJ,DATMSL,ZZSYS,SASYS,LBRSYS,INFORM,IDU1)                TIL01730
      IF(NPHASE.LT.6) GO TO 380                                         TIL01740
      ISIDE=1                                                           TIL01750
      CALL  WPNTRM(KBU,KBUSEG,                                          TIL01760
     1      ITRSEG,IRSTAT,KRU,LRSYST,IRTU,MAXRTU,                       TIL01770
     2      INFORM,LBRSYS)                                              TIL01780
      ISIDE=2                                                           TIL01790
      CALL  WPNTRM(KRU,KRUSEG,                                          TIL01800
     1      ITBSEG,IBSTAT,KBU,LBSYST,IBTU,MAXBTU,                       TIL01810
     2      INFORM,LBRSYS)                                              TIL01820
      ISIDE=1                                                           TIL01830
      CALL  ACMTRM(NBTU,KBU,IBSTAT,BTUXYZ,IBTU,INVBTU,MAXBTU,           TIL01840
     1      KRU,IRSTAT,LRSYST,IRTU,MAXRTU,                              TIL01850
     2      DATMSL)                                                     TIL01860
      ISIDE=2                                                           TIL01870
      CALL  ACMTRM(NRTU,KRU,IRSTAT,RTUXYZ,IRTU,INVRTU,MAXRTU,           TIL01880
     1      KBU,IBSTAT,LBSYST,IBTU,MAXBTU,                              TIL01890
     2      DATMSL)                                                     TIL01900
      ISIDE=1                                                           TIL01910
      CALL  STATUS(NBTU,INVBTU,KBUSEG,IBTU,NEXBTU,KBU,ITBSEG,IBSTAT,    TIL01920
     A     MAXBTU,LBSYST,BGC,INTVLB,NBG,NBU,                            TIL01930
     1      KRUSEG,IRSTAT,KRU,IRTU,ITRSEG,                              TIL01940
     2       INFORM,LBRSYS)                                             TIL01950
      ISIDE=2                                                           TIL01960
      CALL  STATUS(NRTU,INVRTU,KRUSEG,IRTU,NEXRTU,KRU,ITRSEG,IRSTAT,    TIL01970
     A     MAXRTU,LRSYST,RGC,INTVLR,NRG,NRU,                            TIL01980
     1      KBUSEG,IBSTAT,KBU,IBTU,ITBSEG,                              TIL01990
     2       INFORM,LBRSYS)                                             TIL02000
      ISIDE=1                                                           TIL02010
      CALL  USTAT(KBU,IBSTAT,ITBSEG,NBTU,KBUK,                          TIL02020
     1      NRTU,INVRTU,IRTU,                                           TIL02030
     2       INFORM)                                                    TIL02040
```

```
      ISIDE=2
      CALL USTAT(KRU,IRSTAT,ITRSEG,NRTU,KRUK,
     1    NBTU,INVBTU,IBTU,
     2    INFORM)
  380 CONTINUE
      IF(JPRINT.EQ.0) GO TO 390
      JPRINT=0
      TPRINT=TIME+PRSTEP
      IF(TPRINT.GT.PSTOP) TPRINT=100.
      IF(TPRINT.LT.PSTART) TPRINT=PSTART
      IF(IMISC(7).EQ.0) GO TO 390
      ISIDE=1
      CALL SNAP(KBU,NBTU,LPSYST,LBRSYS,KBUSEG,ITBSEG,INFORM,
     1    IBSTAT,IBTU,INVBTU,BTUXYZ,  MAXBTU,NAMBU,NABSS,BXYZ,
     2    RXYZ,RTUXYZ,KRU)
      ISIDE=2
      CALL SNAP(KRU,NRTU,LPSYST,LBRSYS,KRUSEG,ITRSEG,INFORM,
     1    IRSTAT,IRTU,INVRTU,RTUXYZ,  MAXRTU,NAMRU,NARSS,RXYZ,
     2    BXYZ,BTUXYZ,KBU)
      IF(NPHASE.LT.2) RETURN
  390 TLAST=TIME
      TIME=TIME+SSTEP
      IF(TIME.LT.TIMEB) GO TO 3
  395 CONTINUE
      RETURN
      END


      SUBROUTINE HVTGT(KVEC,NWG,INTVLW,WGC,WGCV,
     1    KTU,ITSTAT,KTUK,TXYZ,TREL,IVAL,KHVT,
     2    TIME    )
      DIMENSION TREL(4,1),ITSTAT(1),KTUK(4,1),INTVLW(1),WGC(8,6,1),
     1    WGCV(8,5,1),TXYZ(50,1),KHVT(1)
      DIMENSION LHVT(50)
C
      NHVT=0
      VAL=IVAL
      IF(VAL.LT.1.) GO TO 115
      DO 110 ITGT=1,KTU
      KHV=KHVT(ITGT)/100
      IF(KHV.EQ.0) GO TO 110
      KHV=100
      IF(ITSTAT(ITGT).LT.2) GO TO 110
      IF(VAL.GT.TREL(4,ITGT)) GO TO 110
      IF(MOD(KTUK(2,ITGT)/1000000,10).NE.1) GO TO 110
      KHV=1100
      NHVT=NHVT+1
      LHVT(NHVT)=ITGT
  110 KHVT(ITGT)=KHV
      IF(NHVT.EQ.0) GO TO 115
      DO 114 IG=1,NWG
      LEG=IABS(INTVLW(IG))
      DELT=TIME-WGC(LEG,1,IG)
      WX=WGC(LEG,2,IG)+WGCV(LEG,1,IG)*DELT
      WY=WGC(LEG,3,IG)+WGCV(LEG,2,IG)*DELT                              HVTG0270
      GR2=1.E50                                                         HVTG0280
      DO 113 I=1,NHVT                                                   HVTG0290
      ITGT=LHVT(I)                                                      HVTG0300
      TX=TXYZ(ITGT,2)                                                   HVTG0310
      TY=TXYZ(ITGT,3)                                                   HVTG0320
      DX=TX-WX                                                          HVTG0330
      DY=TY-WY                                                          HVTG0340
      X=DX*DX+DY*DY                                                     HVTG0350
      IF(X.GT.GR2) GO TO 113                                            HVTG0360
      GR2=X                                                             HVTG0370
      IMIN=ITGT                                                         HVTG0380
  113 CONTINUE                                                          HVTG0390
      KHVT(IG)=IMIN+KHVT(IG)                                            HVTG0400
      IF(KVEC.EQ.0) GO TO 114                                           HVTG0410
      IF(INTVLW(IG+18).EQ. 1) GO TO 114                                 HVTG0420
      ITGT=IMIN                                                         HVTG0430
      INTVLW(IG)=10+LEG                                                 HVTG0440
      WGC(1,5,IG)=TXYZ(ITGT,2)                                          HVTG0450
      WGC(1,6,IG)=TXYZ(ITGT,3)                                          HVTG0460
  114 CONTINUE                                                          HVTG0470
  115 KHVT(19)=NHVT+KHVT(19)                                            HVTG0480
      RETURN                                                            HVTG0490
      END                                                               HVTG0500


      SUBROUTINE GCDATA(GC,GCV,NG,INTVAL,XR,YR)                         GCDA0010
      DIMENSION GC(8,6,18),GCV(8,5,18),INTVAL(1)                        GCDA0020
      DATA PI,RAD/3.141593, .0174533/                                   GCDA0030
      INTVAL(37)=XR                                                     GCDA0040
      INTVAL(38)=YR                                                     GCDA0050
      DO 100 I=1,NG                                                     GCDA0060
      INTVAL(I+18)=0                                                    GCDA0070
      INTVAL(I)=7                                                       GCDA0080
      DO 100 J=2,8                                                      GCDA0090
      HV=GC(J,5,I)                                                      GCDA0100
      K=HV                                                              GCDA0110
      H=K                                                               GCDA0120
      K=(HV-H)*1000.+.9                                                 GCDA0130
      V=K                                                               GCDA0140
      H=H*RAD                                                           GCDA0150
      GCV(J,5,I)=H                                                      GCDA0160
      IF(H.LT.PI) H=PI*.5-H                                             GCDA0170
      IF(H.GE.PI) H=PI*2.5-H                                            GCDA0180
      GCV(J,1,I)=V*COS(H)                                               GCDA0190
      GCV(J,2,I)=V*SIN(H)                                               GCDA0200
      GCV(J,3,I)=GC(J,6,I)                                              GCDA0210
      GCV(J,4,I)=V                                                      GCDA0220
  100 CONTINUE                                                          GCDA0230
      RETURN                                                            GCDA0240
      END                                                               GCDA0250
      SUBROUTINE SONARM
CSONAR                                       15     5     8     12
       COMMON/CSONAR/NSON,NSONMX,NBSON,NRSON,LASON,NAMSON( 2,15),
     1    KSONTY(15), SONSYS(12,15)
      COMMON /SONCRV/ IBA,IBB,IDA,IDB,IBOTTM,IACTIV,SRANGE(17),
     1      SMERIT(17)
      IF(NSON.EQ.0) RETURN
      DO 100 I=1,IDB
  100 SRANGE(I)=ALOG(SRANGE(I)*.4934)
      IF(IBOTTM.EQ.0) GO TO 110
      IA=IBA
      IB=IBB
      GO TO 115
  110 IA=IDA
      IB=IDB
  115 DO 250 IS=1,NSON
      IPASS=1
      ICOL=12
      SM=SONSYS(11,IS)-(SONSYS(8,IS)-SONSYS(9,IS))-SONSYS(10,IS)
      GO TO 160
  150 IPASS=2
      ICOL=6
      SM=.5*(SONSYS(1,IS)+SONSYS(5,IS)-(SONSYS(2,IS)-SONSYS(3,IS))
     1                                             -SONSYS(4,IS))
  160 SM=ABS(SM)
      IF(SM.GE.SMERIT(IA)) GO TO 200
      RM=0.1
      GO TO 225
  200 IF(SM.LT.SMERIT(IB)) GO TO 205
      RM=SRANGE(IB)
      GO TO 220
  205 I=IA
  210 I=I+1
      IF(I.GT.IB) GO TO 200
      SMI=SMERIT(I)
      IF(SM.GT.SMI) GO TO 210
      J=I-1
      SMJ=SMERIT(J)
      XS=(SM-SMJ)/(SMI-SMJ)
      RM=SRANGE(J)
      RM=RM+XS*(SRANGE(I)-RM)
  220 RM=EXP(RM)
  225 SONSYS(ICOL,IS)=RM
      IF(IPASS.EQ.1) GO TO 150
  250 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SYSTEM(KPXS,KPLAT,KSYST,NPLAT,NAIR,NSYS,RADM,
     1      LBRSYS,IBRS,LENSYS,NUNIT,KUK,LSYST,IGPAIR,NWU,NWG,ISTAT,
     2      PX1,SAMDAT,WPNDAT,DATMSL,AIRCPT,NVALT,IACTIV,SONSYS,TTRAJ)
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12
      COMMON /BATTRY/ NBATT,MISRAD(2,10)
      DIMENSION KPXS(45,1),KPLAT(1),KSYST(1),KSTEMP(45),BETA(2),
     1     LBETA(2),RADM(36,1),LASB(5),LSAG(5),LJAM(5),LASW(5),LCM(5),
     2     LAA(5),LTB(5),LBRSYS(1),LSYST(1),KJK(4,1),LSAM(10)
      DIMENSION IGPAIR(1),NWU(1),ISTAT(1),LSYS(11),LTYPE(11),
     1     PX1(12,1),WPNDAT(15,1),SAMDAT(20,1),DATMSL(24,1)
     2,    SONSYS(12,1),TTPAJ(50,1)
      DATA LTYPE/3*1,6*2,2*3/
      DATA LSYS/20,12,0,-1,12,22,16,23,0,12,0/
      JPACK(NM,LCOL,LS)=MOD(NM,100)*1000000+(NM/100)*10000+LCOL*100+LS
      NAMADD=0
      IF(NAIR.EQ.82) NAMADD=200
      NSUB=NAIR+1
      NSHIP=NAIR-1
      NAM2=1000000
      NAM3=100000
      NAM4=10000
      DO 100 IPLAT=1,NPLAT
      NAMEPL=KPLAT(IPLAT)
      NAMP=NAMEPL/NAM2
      IF(NAMP.LE.NSUB .AND. NAMP.GE.NSHIP) GO TO 10
      WRITE(N6,6100) IPLAT,NAMEPL
      CALL PAGE
 6100 FORMAT(//6X,'PLATFORM',I6,'  CODE=',I10,' NOT A SHIP,A/C, OR SUB')
      STOP
   10 NSAM=0
      NSAG=0
      NJAM=0
      NASW=0
      NTB=0
      NCM=0
      NAA=0
      NASB=0
      DO 110 ISYS=1,NSYS
  110 KSTEMP(ISYS)=KPXS(ISYS,IPLAT)
      BETA(1)=0.
      BETA(2)=0.
      LBETA(1)=0
      LBETA(2)=0
      IF(NAMP-NAIR) 121,122,123
  121 IPASS=1
      KSYS=NAMADD+651
      GO TO 130
  122 IPASS=2
      KSYS=NAMADD+657
      GO TO 130
  123 IPASS=2
      KSYS=NAMADD+654
  130 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)
      IF(LS.EQ.0) GO TO 140
      LCOL=MOD(KSYST(LS),100)
      IF(LCOL.EQ.0) GO TO 130
      IF(RADM(4,LCOL) .LE.0.) GO TO 130
      MODE=RADM(36,LCOL)+.1
      IF(MODE.GT.1) MODE=2
      B=RADM(1,LCOL)
      IF(B.LE.BETA(MODE)) GO TO 130
      BETA(MODE) =B
      LBETA(MODE)=LCOL*100+LS
      GO TO 130
  140 IF(IPASS.EQ.2) GO TO 150
      IPASS=2
      KSYS=NAMADD+652
      GO TO 130
  150 ICOL=12-IACTIV*6
      IPASS=1
      KSYS=NAMADD+653
      RSONR=0.
      NSONR=0
  152 CALL SYSTST(KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)
      IF(LS.EQ.0) GO TO 155
      LCOL=MOD(KSYST(LS),100)
      IF(LCOL.EQ.0) GO TO 152
      B=SONSYS(ICOL,LCOL)
      IF(RSONR.GE.B) GO TO 152
      RSONR=B
      NSONR=LCOL*100+LS
      GO TO 152
  155 IF(IPASS.EQ.2) GO TO 2
      IPASS=2
      KSYS=NAMADD+655
      GO TO 152
    2 IF(NAMP-NAIR) 20,28,28
   20 KSYS=NAMADD+671
  200 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)
      IF(LS.EQ.0) GO TO 240
      LRS=LS
      LCOL=MOD(KSYST(LS),100)
      IF(LCOL.EQ.0) GO TO 200
      IF(RADM(4,LCOL).LE.0.) GO TO 200
      NRAD=NAMS/NAM4
      DO 210 I=1,NBATT
      LRAD=MISRAD(1,I)
      IF(NRAD.NE.LRAD) GO TO 210
      LMIS=MISRAD(2,I)
      GO TO 215
  210 CONTINUE
      GO TO 200
  215 CALL SYSTST (LMIS,LS,NAMM,KSTEMP,NSYS,NAM4)
      IF(LS.EQ.0) GO TO 200
      JCOL=MOD(KSYST(LS),100)
      IF(JCOL.EQ.0) GO TO 200
      NM=MOD(NAMM,10000)
      IF(NM.EQ.0) GO TO 200
      NR=MOD(NAMS,10000)
      IF(NR.EQ.0) GO TO 200
      NSAM=NSAM+1
      LSAM(NSAM)=JPACK(NR,LCOL,LRS)
      NSAM=NSAM+1
      LSAM(NSAM)=JPACK(NM,JCOL,LS)
      GO TO 200
  240 KSYS=NAMADD+751                                                   SYST1160
      IPASS=1                                                           SYST1170
  250 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)                       SYST1180
      IF(LS.EQ.0) GO TO 260                                             SYST1190
      NM=MOD(NAMS,10000)                                                SYST1200
      IF(NM.EQ.0) GO TO 250                                             SYST1210
      LCOL=MOD(KSYST(LS),100)                                           SYST1220
      IF(LCOL.EQ.0) GO TO 250                                           SYST1230
      NSAG=NSAG+1                                                       SYST1240
      LSAG(NSAG)=JPACK(NM,LCOL,LS)                                      SYST1250
      GO TO 250                                                         SYST1260
  260 IF(IPASS.EQ.2) GO TO 28                                           SYST1270
      IPASS=2                                                           SYST1280
      KSYS=NAMADD+752                                                   SYST1290
      GO TO 250                                                         SYST1300
   28 KSYS=NAMADD+681                                                   SYST1310
  280 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)                       SYST1320
      IF(LS.EQ.0) GO TO 30                                              SYST1330
      NM=MOD(NAMS,10000)                                                SYST1340
      IF(NM.EQ.0) GO TO 280                                             SYST1350
      LCOL=MOD(KSYST(LS),100)                                           SYST1360
      IF(LCOL.EQ.0) GO TO 280                                           SYST1370
      NJAM=NJAM+1                                                       SYST1380
      LJAM(NJAM)=JPACK(0,LCOL,LS)                                       SYST1390
      GO TO 280                                                         SYST1400
   30 IF(NAMP-NAIR) 31,31,40                                            SYST1410
   31 KSYS=NAMADD+731                                                   SYST1420
      IPASS=1                                                           SYST1430
  310 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)                       SYST1440
      IF(LS.EQ.0) GO TO 320                                             SYST1450
      NM=MOD(NAMS,10000)                                                SYST1460
      IF(NM.EQ.0) GO TO 310                                             SYST1470
      LCOL=MOD(KSYST(LS),100)                                           SYST1480
      IF(LCOL.EQ.0) GO TO 310                                           SYST1490
      NASW=NASW+1                                                       SYST1500
      LASW(NASW)=JPACK(NM,LCOL,LS)                                      SYST1510
      GO TO 310                                                         SYST1520
  320 IF(IPASS.EQ.2) GO TO 40                                           SYST1530
      IPASS=2                                                           SYST1540
      KSYS=NAMADD+741                                                   SYST1550
      GO TO 310                                                         SYST1560
   40 IF(NAMP-NAIR) 41,42,41                                            SYST1570
   41 IPASS=3                                                           SYST1580
      KSYS=NAMADD+721                                                   SYST1590
      GO TO 410                                                         SYST1600
   42 IPASS=1                                                           SYST1610
      KSYS=NAMADD+783                                                   SYST1620
      GO TO 410                                                         SYST1630
   43 IPASS=2                                                           SYST1640
      KSYS=NAMADD+784                                                   SYST1650
  410 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)                       SYST1660
      IF(LS.EQ.0) GO TO 44                                              SYST1670
      LCOL=MOD(KSYST(LS),100)                                           SYST1680
      IF(LCOL.EQ.0) GO TO 410                                           SYST1690
      NM=MOD(NAMS,10000)                                                SYST1700
      IF(NM.EQ.0) GO TO 410
      IF(IPASS.EQ.3) GO TO 435
      NASB=NASB+1
      LASB(NASB)=JPACK(NM,LCOL,LS)
      GO TO 410
  435 NTB=NTB+1
      LTB(NTB)=JPACK(NM,LCOL,LS)
      GO TO 410
   44 IF(IPASS-2) 43,41,48
   48 IF(NAMP-NAIR) 51,50,51
   50 KSYS=NAMADD+781
      GO TO 510
   51 KSYS=NAMADD+761
  510 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)
      IF(LS.EQ.0) GO TO 60
      LCOL=MOD(KSYST(LS),100)
      IF(LCOL.EQ.0) GO TO 510
      NM=MOD(NAMS,10000)
      IF(NM.EQ.0) GO TO 510
      NCM=NCM+1
      LCM(NCM)=JPACK(NM,LCOL,LS)
      GO TO 510
   60 IF(NAMP-NAIR) 70,61,70
   61 KSYS=NAMADD+791
      IPASS=1
  610 CALL SYSTST (KSYS,LS,NAMS,KSTEMP,NSYS,NAM3)
      IF(LS.EQ.0) GO TO 62
      LCOL=MOD(KSYST(LS),100)
      IF(LCOL.EQ.0) GO TO 610
      NM=MOD(NAMS,10000)
      IF(NM.EQ.0) GO TO 610
      NAA=NAA+1
      LAA(NAA)=JPACK(NM,LCOL,LS)
      GO TO 610
   62 IF(IPASS.EQ.2) GO TO 70
      IPASS=2
      KSYS=NAMADD+792
      GO TO 610
   70 CONTINUE
      KSHIP=0
      KAIR=0
      KSUB=0
      KJAM=0
      KRADR=0
      KSONR=0
      ISTART=IBRS+1
      IF(NAMP.EQ.NSUB) GO TO 711
      ITYPE=1
      DO 710 JTYPE=1,2
      LCOL=LBETA(JTYPE)
      NM=BETA(JTYPE)
      IF(LCOL.EQ.0) GO TO 710
      KRADR=1
      IBRS=IBRS+1
      LBRSYS(IBRS)=(NM*10000+LCOL)*100+JTYPE*10+ITYPE
  710 CONTINUE                                                          SYST2260
  711 IF(NSONR.EQ.0) GO TO 718                                          SYST2270
      KSONR=1                                                           SYST2280
      NM=RSONR*10.                                                      SYST2290
      IBRS=IBRS+1                                                       SYST2300
      LBRSYS(IBRS)=(NM*10000+NSONR)*100+17                              SYST2310
  718 IF(NCM.EQ.0) GO TO 73                                             SYST2320
      ITYPE=12                                                          SYST2330
      KSHIP=1                                                           SYST2340
      DO 720 I=1,NCM                                                    SYST2350
      IBRS=IBRS+1                                                       SYST2360
      LBRSYS(IBRS)=LCM(I)*100+ITYPE                                     SYST2370
  720 CONTINUE                                                          SYST2380
   73 IF(NASB.EQ.0) GO TO 74                                            SYST2390
      ITYPE=22                                                          SYST2400
      KSHIP=1                                                           SYST2410
      DO 730 I=1,NASB                                                   SYST2420
C                                                                       SYST2430
      IBRS=IBRS+1                                                       SYST2440
      LBRSYS(IBRS)=LASB(I)*100+ITYPE                                    SYST2450
  730 CONTINUE                                                          SYST2460
   74 IF(NAA.EQ.0) GO TO 75                                             SYST2470
      ITYPE=23                                                          SYST2480
      KAIR=1                                                            SYST2490
      DO 740 I=1,NAA                                                    SYST2500
      IBRS=IBRS+1                                                       SYST2510
      LBRSYS(IBRS)=LAA(I)*100+ITYPE                                     SYST2520
  740 CONTINUE                                                          SYST2530
   75 IF(NSAM.EQ.0) GO TO 76                                            SYST2540
      ITYPE=33                                                          SYST2550
      KAIR=1                                                            SYST2560
      DO 750 I=1,NSAM,2                                                 SYST2570
      IBRS=IBRS+1                                                       SYST2580
      LBRSYS(IBRS)=LSAM(I)*100+ITYPE                                    SYST2590
      IBRS=IBRS+1                                                       SYST2600
      LBRSYS(IBRS)=LSAM(I+1)*100+30                                     SYST2610
  750 CONTINUE                                                          SYST2620
   76 IF(NSAG.EQ.0) GO TO 77                                            SYST2630
      IF(NAMP.EQ.NSUB) GO TO 77                                         SYST2640
      ITYPE=43                                                          SYST2650
      KAIR=1                                                            SYST2660
      DO 760 I=1,NSAG                                                   SYST2670
      IBRS=IBRS+1                                                       SYST2680
      LBRSYS(IBRS)=LSAG(I)*100+ITYPE                                    SYST2690
  760 CONTINUE                                                          SYST2700
   77 IF(NASW.EQ.0) GO TO 78                                            SYST2710
      ITYPE=24                                                          SYST2720
      KSUB=1                                                            SYST2730
      DO 770 I=1,NASW                                                   SYST2740
      IBRS=IBRS+1                                                       SYST2750
      LBRSYS(IBRS)=LASW(I)*100+ITYPE                                    SYST2760
  770 CONTINUE                                                          SYST2770
   78 IF(NJAM.EQ.0) GO TO 781                                           SYST2780
      ITYPE=15                                                          SYST2790
      KJAM=1                                                            SYST2800
      DO 780 I=1,NJAM
      IBRS=IBRS+1
      LBRSYS(IBRS)=LJAM(I)*100+ITYPE
  780 CONTINUE
  781 IF(NTB.EQ.0) GO TO 783
      ITYPE=16
      KSHIP=1
      KSUB=1
      DO 782 I=1,NTB
      IBRS=IBRS+1
      LBRSYS(IBRS)=LTB(I)*100+ITYPE
  782 CONTINUE
  783 IF(NAMP.NE.NSHIP) GO TO 785
      KSYS=NAIR
 7830 CALL  SYSTST(KSYS,LS,NAMS,KSTEMP,NSYS,NAM2)
      IF(LS.EQ.0) GO TO 785
      IBRS=IBRS+1
      LBRSYS(IBRS)=MOD(NAMS,100)*1000000+NAMS/10000*100+20
      GO TO 7830
  785 MASK=KRADR+KSONR*10
      MASK=MASK*10+KJAM
C
C-- TEMPORARY MISSION CONSTRAINT ON RED SUBS
      IF(NAMP.EQ.NSUB .AND. NAMADD.GT.0) KSUB=0
C
      MASK=MASK*10+KSUB
      MASK=MASK*10+KAIR
      MASK=MASK*10+KSHIP
C
      IF(NAMEPL/NAM3 .NE. NAMADD+624) GO TO 786
      IF(KRADR.EQ.0) GO TO 786
C
      DO 7850 I=1,10
      IBRS=IBRS+1
 7850 LBRSYS(IBRS)=0
      MASK=(MASK/100)*100+1000011
      GO TO 787
  786 IF(MASK.EQ.0) GO TO 85
  787 IBRS=IBRS+1
      LBRSYS(IBRS)=0
      IBRS=IBRS+1
      LBRSYS(IBRS)=0
      IBRS=IBRS+1
      LBRSYS(IBRS)=MASK*10+MOD(NAMP,10)
   79 IF(IBRS.LT.ISTART) GO TO 85
      LAST=IBRS
      LENGTH=LAST-ISTART+1
      ASSIGN 801 TO NU
      DO 800 IU=1,NUNIT
      IF(NAMEPL.NE.KUK(2,IU)) GO TO 800
      GO TO NU,(801,802)
  801 LSYST(IU)=ISTART*1000000+LAST*100+IPLAT
      ASSIGN 802 TO NU
      GO TO 800
  802 LSYST(IU)=(IBRS+1)*1000000+(IBRS+LENGTH)*100+IPLAT
```

```
      DO 810 I=ISTART,LAST                                              SYST3360
      IBRS=IBRS+1                                                       SYST3370
  810 LBRSYS(IBRS)=LBRSYS(I)                                            SYST3380
      IF(IBRS.LE.LENSYS) GO TO 800                                      SYST3390
      CALL PAGE                                                         SYST3400
      WRITE(N6,6800) IBRS,LENSYS,(LBRSYS(K),K=1,IBRS)                   SYST3410
      STOP                                                              SYST3420
 6800 FORMAT(//6X,'LBRSYS ARRAY LENGTH EXCEEDED',//6X,2I10//            SYST3430
     1       (6X,10I10))                                                SYST3440
  800 CONTINUE                                                          SYST3450
      GO TO 100                                                         SYST3460
   85 DO 850 IU=1,NUNIT                                                 SYST3470
      IF(NAMEPL.NE.KUK(2,IU)) GO TO 850                                 SYST3480
      LSYST(IU)=IPLAT                                                   SYST3490
  850 CONTINUE                                                          SYST3500
  100 CONTINUE                                                          SYST3510
      DO 900 IU=1,NUNIT                                                 SYST3520
      ISTAT(IU)=1                                                       SYST3530
      LS=LSYST(IU)                                                      SYST3540
      ISA=LS/1000000                                                    SYST3550
      IF(ISA.EQ.0) GO TO 900                                            SYST3560
      ISB=MOD(LS/100,10000)                                             SYST3570
      MASK=LBRSYS(ISB)                                                  SYST3580
      ITYPE=MOD(MASK,10)                                                SYST3590
      IF(ITYPE.NE.2) GO TO 900                                          SYST3600
      IVEC=1                                                            SYST3610
      NPL=KUK(2,IU)/10000                                               SYST3620
      ICAR=0                                                            SYST3630
      IF(MOD(MASK/10000000,10) .EQ. 0) IVEC=0                           SYST3640
      IG=KUK(3,IU)                                                      SYST3650
      IGB=IGPAIR(IG)/100                                                SYST3660
      ISS=ISB-1                                                         SYST3670
      DO 855 I=ISA,ISS                                                  SYST3680
      IS=I-1                                                            SYST3690
      IF(LBRSYS(I).EQ.0) GO TO 856                                      SYST3700
  855 CONTINUE                                                          SYST3710
  856 CONTINUE                                                          SYST3720
      IF(IS.EQ.ISB-2) IVEC=0                                            SYST3730
      IPASS=1                                                           SYST3740
      JS=0                                                              SYST3750
  860 IUB=NWU(IG)                                                       SYST3760
      IUA=IUB/100                                                       SYST3770
      IUB=IUA+MOD(IUB,100)-1                                            SYST3780
      ISYS=12                                                           SYST3790
      DO 870 IUI=IUA,IUB                                                SYST3800
      LS=LSYST(IUI)                                                     SYST3810
      ISUA=LS/1000000                                                   SYST3820
      IF(ISUA.EQ.0) GO TO 870                                           SYST3830
      ISUB=MOD(LS/100,10000)                                            SYST3840
      MASK=LBRSYS(ISUB)                                                 SYST3850
      ISUB=ISUB-1                                                       SYST3860
      KW=MOD(MASK,10)                                                   SYST3870
      IF(IVEC.EQ.0)    GO TO 867                                        SYST3880
      IF(MOD(MASK/10000000,10) .EQ. 1) GO TO 867                        SYST3890
      ISI=ISUA-1                                                        SYST3900
  865 ISI=ISI+1
      IF(ISI.GT.ISUB) GO TO 867
      IF(MOD(LBRSYS(ISI),100) .NE. ISYS) GO TO 865
  866 IS=IS+1
      JS=JS+1
      IF(JS.GT.10) GO TO 867
      LBRSYS(IS)=IUI*100+10
  867 IF(ICAR.GT.0) GO TO 870
      IF(KW.NE.1) GO TO 870
      DO 868 ISI=ISUA,ISUB
      KSYS=LBRSYS(ISI)
      IF(KSYS/1000000 .EQ. 0) GO TO 868
      IF(MOD(KSYS,100) .NE. 20) GO TO 868
      IF(MOD(KSYS/100,10000) .NE. NPL) GO TO 868
      ICAR=IUI
      LBRSYS(ISB-2)=(IUI*10000+ISI)*100
      GO TO 870
  868 CONTINUE
  870 CONTINUE
      IF(IPASS.EQ.2) GO TO 900
      IF(IGB.EQ.0) GO TO 900
      DO 875 I=1,NWG
      IF(MOD(IGPAIR(I),100) .NE. IGB) GO TO 875
      IG=I
      IPASS=2
      GO TO 860
  875 CONTINUE
  900 CONTINUE
      DO 905 I=1,NUNIT
  905 LSYST(I+NUNIT)=I
      IUNIT=NUNIT
      IUB=2*NUNIT
      DO 920 IPASS=1,11
      IUA=IUNIT+1
      IF(IUA.GT.IUB) GO TO 918
      KSYS=LSYS(IPASS)
      KTYPE=LTYPE(IPASS)
      DO 912 IU=IUA,IUB
      JU=LSYST(IU)
      LS=LSYST(JU)
      ISB=MOD(LS/100,10000)
      IF(ISB.GT.0) GO TO 907
      IF(KSYS.NE.0) GO TO 912
      IF(MOD(KUK(2,JU)/1000000,10).NE.KTYPE) GO TO 912
  906 IUNIT=IUNIT+1
      K=LSYST(IUNIT)
      LSYST(IUNIT)=JU
      LSYST(IU)=K
      GO TO 912
  907 MASK=LBRSYS(ISB)
      IF(MOD(MASK,10).NE.KTYPE) GO TO 912
      IF(KSYS) 910,906,908
  908 ISA=LS/1000000
      ISB=ISB-2
      DO 909 I=ISA,ISB
```

AD-A048 342 LTV AEROSPACE CORP DALLAS TEX VOUGHT SYSTEMS DIV F/G 15/7
SEATIDE ANALYSIS PROCESS. VOLUME IID. NAVAL ENGAGEMENT MODEL (N--ETC(U)
FEB 75 DAAB09-72-C-0062
UNCLASSIFIED VSD-00.1636-VOL-2D-REV-A NL

2 of 3
ADA048 342

```
      IF(MOD(LBRSYS(I),100).EQ.KSYS) GO TO 906
  909 CONTINUE
      GO TO 912
  910 IF(MASK.GT.10000000) GO TO 906
  912 CONTINUE
  918 LSYST(IUB+IPASS)=0
      IF(IUNIT.LT.IUA) GO TO 920
      LSYST(IUB+IPASS)=IUA*10000+IUNIT
  920 CONTINUE
      DO 940 IU=1,NUNIT
      LS=LSYST(IU)
      ISB=MOD(LS/100,10000)
      IF(ISB.EQ.0) GO TO 940
      MASK=LBRSYS(ISB)
      IF(MASK.LT.10000000) GO TO 925
      LBRSYS(ISB-1)=99999990
      GO TO 940
  925 ISL=ISB-2
      ISA=LS/1000000
      IPLAT=MOD(LS,100)
      NMSHIP=0
      NMAIR=0
      KTYPE=MOD(MASK,10)
      DO 935 IS=ISA,ISL
      JS=LBRSYS(IS)
      KSYS=MOD(JS,100)
      ICOL=MOD(JS/10000,100)
      IF(KSYS.NE.12)    GO TO 928
  926 K=TTRAJ(3,ICOL)*1.1
  927 IF(K.GT.NMSHIP) NMSHIP=K
      GO TO 935
  928 IF(KSYS.NE.16)    GO TO 929
      K=WPNDAT(1,ICOL)*2.5
      IF(KTYPE.EQ.2) K=AIRCPT*PX1(NVALT,IPLAT)*2.
      GO TO 927
  929 IF(KSYS.EQ.22)    GO TO 926
      IF(KSYS.NE.23)    GO TO 931
      K=AIRCPT*PX1(NVALT,IPLAT)*2.
  930 IF(K.GT.NMAIR) NMAIR=K
      GO TO 935
  931 IF(KSYS.NE.30    .AND.    KSYS.NE.43) GO TO 935
      K=SAMDAT(1,ICOL)*3.5
      GO TO 930
  935 CONTINUE
      IF(NMAIR.GT.999) NMAIR=999
      IF(NMSHIP.GT.999) NMSHIP=999
      LBRSYS(ISB-1)=NMAIR*100000+NMSHIP*100+90
  940 CONTINUE
      RETURN
      END


      SUBROUTINE SYSTST (KSYS,LS,NAMSYS,KSTEMP,NSYS,LDIV)
      DIMENSION KSTEMP(1)
      DO 100 IS=1,NSYS                                                  SYST0030
      NAMSYS=KSTEMP(IS)                                                 SYST0040
      IF(NAMSYS/LDIV .NE. KSYS) GO TO 100                               SYST0050
      LS=IS                                                             SYST0060
      KSTEMP(IS)=0                                                      SYST0070
      RETURN                                                            SYST0080
  100 CONTINUE                                                          SYST0090
      LS=0                                                              SYST0100
      RETURN                                                            SYST0110
      END                                                               SYST0120


       SUBROUTINE  ALLXYZ(TIME,IPRINT,N6,IXX)                           AXYZ0010
C  PGM=NEM. EVAN COTTEN.        VER.5   9-7-73           FORTRAN IV  EBCD AXYZ0020
C  TO PUT CURRENT POSITIONS IN ALL UNITS                                AXYZ0030
C      BXYZ(I,1) = TIME,                  FOR RED---RXYZ(I,1) = TIME    AXYZ0040
C              2 = X                                     2) = ETC.      AXYZ0050
C              3 = Y                                                    AXYZ0060
C              4 = Z                      NOTE. TIME AND POSITION ARE   AXYZ0070
C              5 = HEAD, RADIANS                LEFT AS IS WHEN         AXYZ0080
C              6 = VEL , KNOTS                  STATUS = KBUK(4,I)= 0   AXYZ0090
C              7 = VERT.VEL.,KNOTS          OR  STATUS = KRUK(4,I)= 0   AXYZ0100
C**                                                                     AXYZ0110
      COMMON/GNAV/ BGCV(8,5,18),RGCV(8,5,18), INTVLB(38),INTVLR(38)     AXYZ0120
       DIMENSION  XYZ(6)                                                AXYZ0130
       COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                 AXYZ0140
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),      AXYZ0150
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),      AXYZ0160
     3          TTIME,NUMX,                                             AXYZ0170
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),   AXYZ0180
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)    AXYZ0190
       DATA  RAD/0.0174533/                                             AXYZ0200
      DATA       ISW/  54321/                                           AXYZ0210
C**  STATEMENT FUNCTIONS                                                AXYZ0220
       FRSMIN(X)=FLOAT(IFIX(X))+FLOAT(IFIX((X-FLOAT(IFIX(X)))*60.5)/60) AXYZ0230
     1          + FLOAT(MOD(IFIX((X-FLOAT(IFIX(X)))*60.5 ),60)) *.01    AXYZ0240
C**                                                                     AXYZ0250
      ASSIGN 14 TO IBSW                                                 AXYZ0260
      ASSIGN 34 TO IRSW                                                 AXYZ0270
      IF(IXX.EQ.ISW) GO TO 1                                            AXYZ0280
      ASSIGN 15 TO IBSW                                                 AXYZ0290
      ASSIGN 35 TO IRSW                                                 AXYZ0300
C                                                                       AXYZ0310
C                                                                       AXYZ0320
C                                                                       AXYZ0330
C                                                                       AXYZ0340
C                                                                       AXYZ0350
C                                                                       AXYZ0360
 1      IXX = 0                                                         AXYZ0370
  10  IF (NBG.LE.0. OR .KBU.LE.0)  GO TO 31                             AXYZ0380
C**  BLU UNITS                                                          AXYZ0390
       DO 24 KG=1,NBG                                                   AXYZ0400
       KL  = NBU(KG)                                                    AXYZ0410
       KUG = MOD(KL,100)                                                AXYZ0420
      KL   = KL / 100
      KLA = KL + KUG - 1
     GO TO IBSW,(14,15)
   14 CALL GRPMOV(TIME,XYZ,IXY,BGC(1,1,KG),BGCV(1,1,KG),INTVLB(KG))
     IF(INTVLB(KG).LT.0) IXX=10
     GO TO 16
   15  CALL  BLUXYZ(TIME,KG,XYZ,IXY)
      IXX = IXX + IXY*IXY
   16  DO 24 KU=KL,KLA
     IF(KBUK(4,KU).EQ.0) GO TO 24
      BXYZ(KU,1) = TIME
      DO 20 I=2,4
 20   BXYZ(KU,I) = XYZ(I-1) + BREL(I-1,KU)
      DO 22 I=5,7
 22   BXYZ(KU,I) = XYZ(I-1)
 24   CONTINUE
C**  RED UNITS
 31   IF (NRG.LE.0. OR .KRU.LE.0)  GO TO 50
      DO 44 KG=1,NRG
      KL   = NRU(KG)
      KUG = MOD(KL,100)
      KL   = KL / 100
      KLA = KL + KUG - 1
     GO TO IRSW,(34,35)
   34 CALL GRPMOV(TIME,XYZ,IXY,RGC(1,1,KG),RGCV(1,1,KG),INTVLR(KG))
     IF(INTVLR(KG).LT.0) IXX=IXX/10*10+1
     GO TO 36
   35  CALL  REDXYZ(TIME,KG,XYZ,IXY)
      IXX = IXX + IXY*IXY
   36  DO 44 KU=KL,KLA
     IF(KRUK(4,KU).EQ.0) GO TO 44
      RXYZ(KU,1) = TIME
      DO 40 I=2,4
 40   RXYZ(KU,I) = XYZ(I-1) + RREL(I-1,KU)
      DO 42 I=5,7
 42   RXYZ(KU,I) = XYZ(I-1)
 44   CONTINUE
 50   TTIME = TIME
      IF (IPRINT.LE.0)  RETURN
C**        * * *              * * *                 * * *
C**
      ENTRY  PRXYZ(N6)
C  TO PRINT CURRENT POSITION OF ALL UNITS
 52   IF (KBU.LE.0)  GO TO 58
 2000 FORMAT(6X,'BLU UNITS, TOTAL=',I5)
 2002 FORMAT(6X,'RED UNITS, TOTAL=',I5)
 2004 FORMAT( 5X,' UNIT  NAME',10X,'CODE',6X,'TYPE GRUP STAT',6X,
    1 'TIME',5X,'X-NM',6X,'Y-NM',5X,'Z-KFT',5X,'HEAD',6X,'VEL-KT',4X,
    2 'Z-KF/M',4X,'HOURS' )
 2006 FORMAT( 5X,I4,'.  ',2A4,2I10,2I5,F10.2,3F10.3,2F10.2,F10.3,F10.4)
C**  BLU UNITS
      CALL  PAGE
      WRITE(N6,2000) KBU
      WRITE(N6,2004)
      DO 56 J=1,KBU
      TIM  = HRSMIN(BXYZ(J,1))
      ZKFT = BXYZ(J,4) * 6.080
      HEAD = BXYZ(J,5) / RAD
      ZFPM = BXYZ(J,7) * 6.080 / 60.
      WRITE(N6,2006)    J,(NAMBU( J,I),I=1,2),(KBUK(I, J),I=1,4),
     1  TIM,(BXYZ(J,I),I=2,3),ZKFT,HEAD,BXYZ(J,6),ZFPM,BXYZ(J,1)
 56   CONTINUE
C**  RED UNITS
 58   IF (KRU.LE.0)  GO TO 64
      CALL  PAGE
      WRITE(N6,2002) KRU
      WRITE(N6,2004)
      DO 62 J=1,KRU
      TIM  = HRSMIN(RXYZ(J,1))
      ZKFT = RXYZ(J,4) * 6.080
      HEAD = RXYZ(J,5) / RAD
      ZFPM = RXYZ(J,7) * 6.080 / 60.
      WRITE(N6,2006)    J,(NAMRU( J,I),I=1,2),(KRUK(I, J),I=1,4),
     1  TIM,(RXYZ(J,I),I=2,3),ZKFT,HEAD,RXYZ(J,6),ZFPM,RXYZ(J,1)
 62   CONTINUE
 64   RETURN
C**          * * *          * * *               * * *
C
      ENTRY  BUNXYZ(TIME,KU,UXYZ,IPRINT,NG,IXX)
C  TO FIND POSITION OF A SINGLE UNIT IN A GROUP - BLU
C    UXYZ(1) =  TIME
C          2  =  X
C          3  =  Y
C          4  =  Z                         NOTE. TIME AND POSITION ARE
C          5  =  HEAD, RADIANS(PRINT DEG)      LEFT AS IS WHEN
C          6  =  VEL, KNOTS                     STATUS = KBUK(4,I) = 0
C          7  =  VERT.VEL.,KNOTS             OR  STATUS = KRUK(4,I) = 0
C    KU   = LINE NO. OF UNIT IF  KU.LT.1000
C         = UNIT CODE, OTHERWISE, E.G. 2020100
C**
      DIMENSION  UXYZ(7)
 2010 FORMAT(1H0,5X,'UNIT XYZ FOR UNIT NO.=',I5)
 2012 FORMAT(1H0,5X'NO UNIT NO.=',I10)
 100  IF (KBU.LE.0)  GO TO 1021
      KTPAN = 1
      IXX = 0
      K   = KU
      IF (KU.LT.1000)  GO TO 104
      DO 102 J = 1, KBU
      IF (KBUK(1,J).EQ.KU)  GO TO 103
 102  CONTINUE
 1021 IXX = 2
      WRITE(N6,2012) KU
      RETURN
 103  K   = J
 104  KG  = MOD(KBUK(3,K),100)
     CALL BLUXYZ(TIME,KG,XYZ,IXY)
      IXX = IXY
      GO TO 118
C**          * * *          * * *               * * *
```

```
C                                                                       AXYZ1530
      ENTRY  RUNXYZ(TIME,KU,UXYZ,IPRINT,NG,IXX)                         AXYZ1540
C  TO FIND POSITION OF A SINGLE UNIT IN A GROUP - RED                   AXYZ1550
 110  IF (KRU.LE.0)  GO TO 1021                                         AXYZ1560
      KTRAN = 2                                                         AXYZ1570
      IXX = 0                                                           AXYZ1580
      K   = KU                                                          AXYZ1590
      IF (KU.LT.1000)  GO TO 114                                        AXYZ1600
      DO 112 J=1,KRU                                                    AXYZ1610
      IF (KRUK(1,J).EQ.KU)  GO TO 113                                   AXYZ1620
 112  CONTINUE                                                          AXYZ1630
      GO TO 1021                                                        AXYZ1640
 113  K   = J                                                           AXYZ1650
 114  KG  = MOD(KRUK(3,K),100)                                          AXYZ1660
      CALL REDXYZ(TIME,KG,XYZ,IXY)                                      AXYZ1670
      IXX = IXY                                                         AXYZ1680
 118  UXYZ(1) = TIME                                                    AXYZ1690
      IF (KTRAN.EQ.2)  GO TO 122                                        AXYZ1700
      DO 120 J=2,4                                                      AXYZ1710
 120  UXYZ(J) = XYZ(J-1) + BREL(J-1,K )                                 AXYZ1720
     GO TO 126                                                          AXYZ1730
 122  DO 124 J=2,4                                                      AXYZ1740
 124  UXYZ(J) = XYZ(J-1) + RREL(J-1,K )                                 AXYZ1750
 126  DO 128 J=5,7                                                      AXYZ1760
 128  UXYZ(J) = XYZ(J-1)                                                AXYZ1770
      IF (IPRINT.LE.0)  GO TO 134                                       AXYZ1780
      TIM  = HRSMIN(UXYZ(1))                                            AXYZ1790
      ZKFT = UXYZ(4) * 6.080                                            AXYZ1800
      HEAD = UXYZ(5) / RAD                                              AXYZ1810
      ZFPM = UXYZ(7) * 6.080 / 60.                                      AXYZ1820
      WRITE(N6,2010) KU                                                 AXYZ1830
      WRITE(N6,2004)                                                    AXYZ1840
      IF (KTRAN.EQ.2)  GO TO 132                                        AXYZ1850
C**  PLU                                                                AXYZ1860
      WRITE(N6,2006)  K ,(NAMBU(K ,I),I=1,2),(KBUK(I,K ),I=1,4),        AXYZ1870
     1  TIM,(UXYZ(  I),I=2,3),ZKFT,HEAD,UXYZ(  6),ZFPM,UXYZ(  1)        AXYZ1880
      GO TO 134                                                         AXYZ1890
C**  RED                                                                AXYZ1900
 132  WRITE(N6,2006)  K ,(NAMRU(K ,I),I=1,2),(KRUK(I,K ),I=1,4),        AXYZ1910
     1  TIM,(UXYZ(  I),I=2,3),ZKFT,HEAD,UXYZ(  6),ZFPM,UXYZ(  1)        AXYZ1920
 134  RETURN                                                            AXYZ1930
      END                                                               AXYZ1940


      SUBROUTINE BLUXYZ(TIME,KG,XYZ,IXX)                                XYZ 0010
C  PGM=NXX.                VER.2     9-7-73     FORTRAN IV     EBCD     XYZ 0020
C  TO INTERPOLATE FOR POSITION, VECTOR, AND ZDOT  AT TIME GIVEN         XYZ 0030
C     AND FOR GROUP KG                                                  XYZ 0040
C     XYZ(1) = X, NM                                                    XYZ 0050
C     XYZ(2) = Y, NM                                                    XYZ 0060
C     XYZ(3) = Z, NM                                                    XYZ 0070
C     XYZ(4) = HEADING, RADIANS                                         XYZ 0080
C     XYZ(5) = VELOCITY, KNOTS                                          XYZ 0090
C     XYZ(6) = ZDOT (VERTICAL VELOCITY), KNOTS                          XYZ 0100
```

```
C
C     IXX     =  0,IF O.K.
C             = -1,IF TIME TOO EARLY.  RETURN EARLIEST POSITION
C             = +1,IF TIME TOO LATE.   RETURN LAST POSITICN
C**
      DIMENSION  XYZ(6), A(8,6), H(8), V(8)
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,
     1    NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),
     2    NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),
     3         TTIME,NUMX,
     4    KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),
     5    KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)
       DATA RAD,PI/.0174533,3.14159/
    1 KTRAN = 1
      DO 10 I = 2, 8
      DO 10 J = 1, 6
   10 A(I,J) = BGC(I,J,KG)
      GO TO 30
C**                 * * *               * * *                * * *
C
      ENTRY         REDXYZ(TIME,KG,XYZ,IXX)
   11 KTRAN = 2
      DO 20 I = 2, 8
      DO 20 J = 1, 6
   20 A(I,J) = RGC(I,J,KG)
   30 IXX = C
      DO 40 I = 2, 8
      H(I) = AINT(A(I,5))
      J = (A(I,5) - H(I)) * 1000. + .9
      V(I) = J
   40 H(I) = H(I) * RAD
      IF (TIME - A(2,1)) 50, 60, 60
   50 IXX = -1
      I = 2
      GO TO 80
   60 DO 70 I = 2, 8
      IF (ABS(A(I,1)-TIME).LE..0001)  GO TO 80
      IF (A(I,1) - TIME) 70, 80, 90
   70 CONTINUE
      IXX = 1
      I = 8
   80 XYZ(1) = A(I,2)
      XYZ(2) = A(I,3)
      XYZ(3) = A(I,4)
      XYZ(4) = H(I)
      XYZ(5) = V(I)
      XYZ(6) = A(I,6)
      GO TO 100
   90 R = (TIME-A(I-1,1)) / (A(I,1)-A(I-1,1))
      XYZ(1) = (A(I,2)-A(I-1,2)) * R + A(I-1,2)
      XYZ(2) = (A(I,3)-A(I-1,3)) * R + A(I-1,3)
      XYZ(3) = (A(I,4)-A(I-1,4)) * R + A(I-1,4)
      XYZ(4) = H(I-1)
      XYZ(5) = V(I-1)
      XYZ(6) = A(I-1,6)
```

```
 100  RETURN                                                           XYZ 0660
      END                                                              XYZ 0670


      SUBROUTINE GRPMOV(TIME,XYZ,IXX,GC,GCV,LAST)                      GRPM0010
      DIMENSION XYZ(1), GC(8,6), GCV(8,5)                              GRPM0020
      DATA PI /3.141593/                                               GRPM0030
      IP=0                                                             GRPM0040
      IXX=0                                                            GRPM0050
      IS=0                                                             GRPM0060
      L=IABS(LAST)                                                     GRPM0070
    1 IF(L-8) 5,10,2                                                   GRPM0080
    2 IP=1                                                             GRPM0090
      L=MOD(L,10)                                                      GRPM0100
      GO TO 1                                                          GRPM0110
    5 IF(L.EQ.1) GO TO 935                                             GRPM0120
      L=L+1                                                            GRPM0130
   10 DELT=TIME-GC(L,1)                                                GRPM0140
      IF(ABS(DELT) .LT.   .0002) GO TO 80                              GRPM0150
      IF(DELT) 90,80,70                                                GRPM0160
   70 IF(IS) 94,72,73                                                  GRPM0170
   72 IS=1                                                             GRPM0180
   73 IF(L.EQ.8) GO TO 78                                              GRPM0190
      L=L+1                                                            GRPM0200
      GO TO 10                                                         GRPM0210
   90 IF(IS) 91,92,93                                                  GRPM0220
   92 IS=-1                                                            GRPM0230
   91 IF(L.EQ.2) GO TO 79                                              GRPM0240
      L=L-1                                                            GRPM0250
      GO TO 10                                                         GRPM0260
   78 IXX=1                                                            GRPM0270
      GO TO 80                                                         GRPM0280
   79 IXX=-1                                                           GRPM0290
   80 DO 85 I=1,3                                                      GRPM0300
   85 XYZ(I)=GC(L,I+1)                                                 GRPM0310
      GO TO 100                                                        GRPM0320
   93 L=L-1                                                            GRPM0330
  935 DELT=TIME-GC(L,1)                                                GRPM0340
   94 DO 95 I=1,3                                                      GRPM0350
   95 XYZ(I)=GC(L,I+1)+DELT*GCV(L,I)                                   GRPM0360
  100 IF(IP.GT.0) GO TO 200                                            GRPM0370
  102 DO 110 I=4,6                                                     GRPM0380
      K=9-I                                                            GRPM0390
  110 XYZ(I)=GCV(L,K)                                                  GRPM0400
      IF(GCV(L,3).NE.0.) LAST=100                                      GRPM0410
      IF(IABS(LAST).NE.L) L=-L                                         GRPM0420
      LAST=L                                                           GRPM0430
      RETURN                                                           GRPM0440
  200 X=GC(1,5)                                                        GRPM0450
      Y=GC(1,6)                                                        GRPM0460
      DX=X-XYZ(1)                                                      GRPM0470
      DY=Y-XYZ(2)                                                      GRPM0480
      GC(1,2)=X                                                        GRPM0490
      GC(1,3)=Y                                                        GRPM0500
      GC(1,4)=XYZ(3)                                                    GRPM0510
      GR=SQRT(DY*DY+DX*DX)                                              GRPM0520
      V=GCV(L,4)                                                        GRPM0530
      DELT=GR/V                                                         GRPM0540
      GCV(1,1)=V*DX/GR                                                  GRPM0550
      GCV(1,2)=V*DY/GR                                                  GRPM0560
      GCV(1,3)=0.                                                       GRPM0570
      GCV(1,4)=V                                                        GRPM0580
      HEAD=PI*.5-ATAN2(DY,DX)                                           GRPM0590
      IF(HEAD.LT.0.) HEAD=HEAD+2.*PI                                    GRPM0600
      GCV(1,5)=HEAD                                                     GRPM0610
      GC(1,1)=TIME+DELT                                                 GRPM0620
      L=1                                                               GRPM0630
      GO TO 102                                                         GRPM0640
      END                                                               GRPM0650


      SUBROUTINE TUXYZ(NWTU,IWTU,INVWTU,TUWXYZ,MAXWTU,                  TUXY0010
     A      KWUK,INTVLW,WXYZ,WGCV,                                      TUXY0020
     1      KTU,INTVLT,KTUK,TXYZ,TGCV,                                  TUXY0030
     2      TTRAJ,IVEC,ITEM,IDU1)                                       TUXY0040
      COMMON/ECONST/ VAR(15),IMISC(35)                                  TUXY0050
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP         TUXY0060
      DIMENSION TUWXYZ(8,1),INVWTU(1),KTUK(4,1),IWTU(1),INTVLT(1),      TUXY0070
     1      TXYZ(IDU1,7),TGCV(8,5,1),TUTXYZ(8,1),TTRAJ(50,1)            TUXY0080
     2,     KWUK(4,1),INTVLW(1),WXYZ(IDU1,1),WGCV(8,5,1)                TUXY0090
      EQUIVALENCE(IMISC(3),NLPR)                                        TUXY0100
      EQUIVALENCE (VAR(8),AIRCPT)                                       TUXY0110
      DATA N6/6/                                                        TUXY0120
      NAMELIST/LOSTGT/ TIME,IW,NAME,ITGT                                TUXY0130
      ITEM=0                                                            TUXY0140
      ASSIGN 120 TO ISWT                                                TUXY0150
      ASSIGN 102 TO JSWT                                                TUXY0160
      ASSIGN 101 TO KSWT                                                TUXY0170
      GO TO 1                                                           TUXY0180
      ENTRY TU2XYZ(NWTU,IWTU,INVWTU,TUWXYZ,                             TUXY0190
     1      KTU,INTVLT,TUTXYZ,                                          TUXY0200
     2      IVEC,IDU1)                                                  TUXY0210
      ASSIGN 10 TO ISWT                                                 TUXY0220
      ASSIGN 103 TO JSWT                                                TUXY0230
      ASSIGN 150 TO KSWT                                                TUXY0240
    1 IVEC=0                                                            TUXY0250
      IF(NWTU.LE.0) RETURN                                              TUXY0260
      DO 150 IW=1,NWTU                                                  TUXY0270
      KV=IABS(INVWTU(IW))                                               TUXY0280
      IF(KV.EQ.0) GO TO 150                                             TUXY0290
      IWT=IWTU(IW)                                                      TUXY0300
      ITGT=MOD(IWT/100000,1000)                                         TUXY0310
      JTGT=ITGT                                                         TUXY0320
      IF(ITGT.GT.KTU) GO TO ISWT,(120,10)                               TUXY0330
      GO TO KSWT,(150,101)                                              TUXY0340
   10 ITGT=ITGT-KTU                                                     TUXY0350
  101 NAME=MOD(IWT,10)                                                  TUXY0360
      WX=TUWXYZ(1,IW)                                                   TUXY0370
      WY=TUWXYZ(2,IW)                                                   TUXY0380
      WZ=TUWXYZ(3,IW)                                                   TUXY0390
      TVEC=TUWXYZ(8,IW)                                                 TUXY0400
      GO TO JSWT,(102,103)                                              TUXY0410
  102 IF(ITGT.EQ.0) GO TO 1025                                          TUXY0420
      ITG=KTUK(3,ITGT)                                                  TUXY0430
      LEGT=INTVLT(ITG)                                                  TUXY0440
      IF(LEGT.GT.0 .AND. TVEC.GT.TIME) GO TO 113                        TUXY0450
      LEG=IABS(LEGT)                                                    TUXY0460
      TX=TXYZ(ITGT,2)                                                   TUXY0470
      TY=TXYZ(ITGT,3)                                                   TUXY0480
      TZ=TXYZ(ITGT,4)                                                   TUXY0490
      TVX=TGCV(LEG,1,ITG)                                               TUXY0500
      TVY=TGCV(LEG,2,ITG)                                               TUXY0510
      TVZ=TGCV(LEG,3,ITG)                                               TUXY0520
      GO TO 104                                                         TUXY0530
 1025 ITGT=MOD(IWTU(MAXWTU+IW),10000)                                   TUXY0540
      IF(ITGT.GT.100) ITGT=ITGT/100                                     TUXY0550
      ITG=KWUK(3,ITGT)                                                  TUXY0560
      LEGT=INTVLW(ITG)                                                  TUXY0570
      IF(LEGT.GT.0 .AND. TVEC.GT.TIME) GO TO 113                        TUXY0580
      LEG=IABS(LEGT)                                                    TUXY0590
      TX=WXYZ(ITGT,2)                                                   TUXY0600
      TY=WXYZ(ITGT,3)                                                   TUXY0610
      TZ=WXYZ(ITGT,4)                                                   TUXY0620
      TVX=WGCV(LEG,1,ITG)                                               TUXY0630
      TVY=WGCV(LEG,2,ITG)                                               TUXY0640
      TVZ=WGCV(LEG,3,ITG)                                               TUXY0650
      GO TO 104                                                         TUXY0660
  103 LEGT=INTVLT(ITGT)                                                 TUXY0670
      IF(LEGT.GT.0 .AND. TVEC.GT.TIME) GO TO 113                        TUXY0680
      TX=TUTXYZ(1,ITGT)                                                 TUXY0690
      TY=TUTXYZ(2,ITGT)                                                 TUXY0700
      TZ=TUTXYZ(3,ITGT)                                                 TUXY0710
      TVX=TUTXYZ(4,ITGT)                                                TUXY0720
      TVY=TUTXYZ(5,ITGT)                                                TUXY0730
      TVZ=TUTXYZ(6,ITGT)                                                TUXY0740
  104 NV=MOD(KV/10,10)                                                  TUXY0750
      IV=MOD(KV,10)                                                     TUXY0760
      IF(IV.NE.0) GO TO 106                                             TUXY0770
      IF(TVEC.GE.TIME) GO TO 105                                        TUXY0780
      IF(NV.GT.0) GO TO 140                                             TUXY0790
      NV=1                                                              TUXY0800
  105 WVX=TUWXYZ(4,IW)                                                  TUXY0810
      WVY=TUWXYZ(5,IW)                                                  TUXY0820
      WV=SQRT(WVX*WVX+WVY*WVY)                                          TUXY0830
      GO TO 108                                                         TUXY0840
  106 NTRAJ=MOD(IWT/10,100)                                             TUXY0850
      IOFF=(IV-1)*25                                                    TUXY0860
      LV=TTRAJ(IOFF+2,NTRAJ)+.01                                        TUXY0870
      LADJ=MOD(LV,100)                                                  TUXY0880
      LV=LV/100                                                         TUXY0890
      IF(TVEC.GT.TIME) GO TO 107                                        TUXY0900
      NV=NV+1                                                           TUXY0910
      IF(NV.GT.LV) GO TO 140                                            TUXY0920
      IF(NAME.NE.4) GO TO 107
      IF(NV.NE.LADJ+1) GO TO 107
      IWTU(MAXWTU+IW)=100000000+MOD(IWTU(MAXWTU+IW),100000000)
  107 NOFF=(NV-1)*4+IOFF+4
      WV=TTRAJ(NOFF+3,NTRAJ)
      IF(WV.LT.0.) GO TO 140
  108 NS=MOD(KV/100,1000)
      S=NS
      IVEC=1
      CALL  INCEPT(TX,TY,TVX,TVY,WX,WY,WV,WVX,WVY,DELTI)
      IF(DELTI.LT.0.) GO TO 1080
      IF(DELTI.LT.AIRCPT) GO TO 1083
      IF(IV.NE.0 .OR. JTGT.EQ.0) GO TO 1083
 1080 IF(NLPR.GT.0) WRITE(N6,LOSTGT)
C
      TUWXYZ(7,IW)=-100.
      GO TO 150
C
C
C
 1083 TINCPT=TIME+DELTI
 1084 TUWXYZ(4,IW)=WVX
      TUWXYZ(5,IW)=WVY
      TUWXYZ(7,IW)=TINCPT-TSTEP
      INVWTU(IW)=-(KV/100*100+NV*10+IV)
      IF(IV.EQ.0) GO TO 110
      IF(NV.EQ.LV) GO TO 109
      RTEST=TTRAJ(NOFF+6,NTRAJ)
      N=NV+1
      HN=TTRAJ(NOFF+5,NTRAJ)
      IF(N.NE.LADJ) GO TO 1085
      HNP1=TTRAJ(NOFF+9,NTRAJ)
      RNP1=TTRAJ(NOFF+10,NTRAJ)
      DR=RTEST-RNP1
      HN=HNP1+(HN-HNP1)*(DR-S)/DR
 1085 IF(NV.LT.LADJ) RTEST=RTEST-S
10855 DX=TX-WX
      DY=TY-WY
      VX=TVX-WVX
      VY=TVY-WVY
      CALL  GPTEST(DX,DY,VX,VY,RTEST,TINT,TSTART,TSTOP)
      DELTI=TSTART-TIME
 1086 WVZ=0.
      IF(HN.LT..0033) HN=.0033
      IF(DELTI.GT.0.) WVZ=(HN-WZ)/DELTI
      TUWXYZ(6,IW)=WVZ
      TUWXYZ(8,IW)=TSTART-TSTEP
      GO TO 114
  109 IV=0
      INVWTU(IW)=-(KV/100000*100000+NV*10+IV)
      NS=0
  110 HN=TZ+TVZ*DELTI
      IF(NS.EQ.0) GO TO 112
      RTEST=S*.1
      GO TO 10855
```

```
112   TSTART=TINCPT                                                     TUXY1480
      CO TO 1086                                                        TUXY1490
113   WVX=TUWXYZ(4,IW)                                                  TUXY1500
      WVY=TUWXYZ(5,IW)                                                  TUXY1510
      WVZ=TUWXYZ(6,IW)                                                  TUXY1520
      IF(ABS(WVZ).GT..01) KV=-KV                                        TUXY1530
      INVWTU(IW)=KV                                                     TUXY1540
114   TUWXYZ(1,IW)=WX+WVX*TSTEP                                         TUXY1550
      TUWXYZ(2,IW)=WY+WVY*TSTEP                                         TUXY1560
      TUWXYZ(3,IW)=WZ+WVZ*TSTEP                                         TUXY1570
      GO TO 150                                                         TUXY1580
120   ITEM=ITEM+1                                                       TUXY1590
      GO TO 150                                                         TUXY1600
140   INVWTU(IW)=KV                                                     TUXY1610
      TUWXYZ(7,IW)=TIME                                                 TUXY1620
150   CONTINUE                                                          TUXY1630
      RETURN                                                            TUXY1640
      END                                                               TUXY1650


      SUBROUTINE GRTEST(DXO,DYO,VX,VY,RTEST,TINT,TSTART,TSTOP)          GRTE0010
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP         GRTE0020
      DGRA=VX*DXO+VY*DYO                                                GRTE0030
      DGRB=VX*VX+VY*VY                                                  GRTE0040
      GRO2=DXO*DXO+DYO*DYO                                              GRTE0050
      RTEST2=RTEST*RTEST                                                GRTE0060
      TWOA=2.*DGRB                                                      GRTE0070
      IF(TWOA.EQ.0.) GO TO 200                                          GRTE0080
      B=2.*DGRA                                                         GRTE0090
      C=GRO2-RTEST2                                                     GRTE0100
      RADCAL=B*B-2.*TWOA*C                                              GRTE0110
      IF(RADCAL.LE.0.) GO TO 205                                        GRTE0120
      RADCAL=SQRT(RADCAL)                                               GRTE0130
      TSTART=-(B+RADCAL)/TWOA+TIME                                      GRTE0140
      IF(TSTART.GT.TIMAX ) GO TO 205                                    GRTE0150
      TSTOP=(RADCAL-B)/TWOA+TIME                                        GRTE0160
      IF(TSTOP.LT.TIME ) GO TO 205                                      GRTE0170
      IF(TSTART.LT.TIME ) TSTART=TIME                                   GRTE0180
      IF(TSTOP.GT.TIMAX ) TSTOP=TIMAX                                   GRTE0190
201   TINT=TSTOP-TSTART                                                 GRTE0200
      RETURN                                                            GRTE0210
200   IF(GRO2.GT.RTEST2) GO TO 205                                      GRTE0220
      TSTART=TIME                                                       GRTE0230
      TSTOP=TIMAX                                                       GRTE0240
      CO TO 201                                                         GRTE0250
205   TINT=0.                                                           GRTE0260
      TSTART=TIMAX                                                      GRTE0270
      TSTOP=TIME-TSTEP                                                  GRTE0280
      RETURN                                                            GRTE0290
      END                                                               GRTE0300


      SUBROUTINE GUID(TH,PSI,VEL,IG,IE,IL,PLP,PLV,SXT,SYT,CEP)          GUID0010
```

```
C   PGM=NXX(NEM).  EVAN COTTEN    8-1-73               FORTRAN IV     EBCD   GUID0020
C   MIDCOURSE GUIDANCE ERRORS FOR,                                           GUID0030
C     IC    = 1,  ALL INERTIAL.     =2,AIRSPEED.      =3,DOPPLER.            GUID0040
C     NEEDS  BLOCK DATA IN  COMMON/CGUID/                                    GUID0050
C     TH      = FLIGHT TIME, HRS                                             GUID0060
C     PSI     = HEADING, RADIANS                                             GUID0070
C     VEL     = KNOTS                                                        GUID0080
C     IC      =1,ALL INERTIAN        =2,AIRSPEED     =3,DOPPLER              GUID0090
C     IE      =1,FOR 1970-75         =2,FOR 1975-80                          GUID0100
C     IL      =1,FOR LAND LAUNCH     =2,FOR SEA LAUNCH        =3,AIR LAUNCH  GUID0110
C     PLP     = 0.,FOR SMALLEST POSITION ERROR FOR PLATFORM TYPE             GUID0120
C             = 1.,FOR LARGEST                                               GUID0130
C     PLV     = 0.,FOR SMALLEST LAUCH VEL ERRORS FOR PLATFORM TYPE           GUID0140
C             = 1.,FOR LARGEST                                               GUID0150
C     RETURNS,              SXT,SYT=  STD.DEV. IN X & Y, NM.                 GUID0160
C                           CEP =      CIRCULAR PROBABLE ERROR,, NM          GUID0170
      COMMON/CGUID/ SDXP(2),SDYP(2),SKXP(2),SKYP(2),SEXP(2),SEYP(2),         GUID0180
     1SKX(2),SKY(2),SDRXT(2,3),SDRYT(2,3),SDRXDT(2,3),SDRYDT(2,3),THETA GUID0190
      DIMENSION A(5),CX(7),CY(6),SX(7),SY(6)                                 GUID0200
      DATA PI,G,R,WE,A,C/3.14159265,32.2,2.0926388E7,7.2921152E-5,           GUID0210
     10.675617,-0.115956,1.65078,-1.54296,0.509798,1.296E6/                  GUID0220
      SPS = SIN(PSI)                                                         GUID0230
      CPS = COS(PSI)                                                         GUID0240
      VX = VEL * SPS * 1.6878                                                GUID0250
      VY = VEL * CPS * 1.6878                                                GUID0260
      T = TH * 3600.                                                         GUID0270
      SDRX = SDRXT(1,IL) + PLP*(SDRXT(2,IL)-SDRXT(1,IL))                     GUID0280
      SDRY = SDRYT(1,IL) + PLP*(SDRYT(2,IL)-SDRYT(1,IL))                     GUID0290
      GO TO (100, 200, 300), IG                                              GUID0300
  100 THETAR = THETA * PI / 180.                                             GUID0310
      SDRXD = SDRXDT(1,IL) + PLV*(SDRXDT(2,IL)-SDRXDT(1,IL))                 GUID0320
      SDRYD = SDRYDT(1,IL) + PLV*(SDRYDT(2,IL)-SDRYDT(1,IL))                 GUID0330
      WS = SQRT(G/R)                                                         GUID0340
      OWS = 1. / WS                                                          GUID0350
      RHOZ = VX / R * TAN(THETAR)                                            GUID0360
      WEY = WE * COS(THETAR)                                                 GUID0370
      SPZ = SQRT((SEXP(IE)/WEY*2./C*PI)**2 + (SDRYD/(R*WEY))**2)             GUID0380
      CX(2) = SKXP(IE) * VX / 100. * OWS                                     GUID0390
      CY(2) = SKYP(IE) * VY / 100. * OWS                                     GUID0400
      CX(3) = SDRXD * OWS                                                    GUID0410
      CY(3) = SDRYD * OWS                                                    GUID0420
      CX(4) = SKY(IE) / 100. * VX                                            GUID0430
      CY(4) = SKX(IE) / 100. * VY                                            GUID0440
      CX(5) = SDXP(IE) * RHOZ * R                                            GUID0450
      CY(5) = SDYP(IE) * RHOZ * R                                            GUID0460
      CX(6) = SEYP(IE) * PI / C * R * 2.                                     GUID0470
      CY(6) = SPZ * VX                                                       GUID0480
      CX(7) = SPZ * VY                                                       GUID0490
      SWST = SIN(WS*T)                                                       GUID0500
      TM = T - OWS * SWST                                                    GUID0510
      SX(1) = SDRX                                                           GUID0520
      SY(1) = SDRY                                                           GUID0530
      DO 120 I = 2, 3                                                        GUID0540
      SX(I) = CX(I) * SWST                                                   GUID0550
  120 SY(I) = CY(I) * SWST                                                   GUID0560
```

```
      DO 140 I = 4, 6
      SX(I) = CX(I) * TM
  140 SY(I) = CY(I) * TM
      SX(7) = CX(7) * T
      SY(6) = CY(6) * T
      SXT = SX(7) * SX(7)
      SYT = 0.
      DO 160 I = 1, 6
      SXT = SXT + SX(I)*SX(I)
  160 SYT = SYT + SY(I)*SY(I)
      GO TO 400
  200 CA = 0.02
      CC = 0.02
      GO TO 320
  300 CA = 0.006
      CC = 0.012
  320 VGT = SQRT(VX**2 + VY**2) * T
      AT = CA * VGT
      CT = CC * VGT
      SXT = (AT*SPS)**2 + (CT*CPS)**2 + SDRX**2
      SYT = (AT*CPS)**2 + (CT*SPS)**2 + SDRY**2
  400 SXT = SQRT(SXT) / 6076.1
      SYT = SQRT(SYT) / 6076.1
      IF (SYT - SXT) 420, 420, 440
  420 RA = SYT / SXT
      S = SXT
      GO TO 460
  440 RA = SXT / SYT
      S = SYT
  460 CEP = A(5) * RA
      DO 480 I = 1, 3
  480 CEP = (A(-I+5) + CEP) * RA
      CEP = (A(1) + CEP) * S
      RETURN
      END


      SUBROUTINE INCEPT(XT,YT,VXT,VYT,XI,YI,VI,VXI,VYI,DELTI)
      DELTI=-1.
      DXO=XT-XI
      DYO=YT-YI
      IF(DXO.NE.0.) GO TO 100
      IF(DYO.NE.0.) GO TO 110
      VXI=0.
      VYI=0.
      DELTI=0.
   10 RETURN
  100 IF(DYO.NE.0.) GO TO 150
      IF(VI.LT.ABS(VYT)) RETURN
      VYI=VYT
      VXI=SQRT(VI*VI-VYI*VYI)
      ITRY=0
  101 VX=VXT-VXI
      IF(VX.NE.0.) DELTI=-DXO/VX
      IF(DELTI.GE.0.) RETURN                                            INCE0180
      IF(ITRY.GT.0) RETURN                                              INCE0190
      ITRY=1                                                            INCE0200
      VXI=-VXI                                                          INCE0210
      GO TO 101                                                         INCE0220
  110 IF(VI.LT.ABS(VXT)) RETURN                                         INCE0230
      VXI=VXT                                                           INCE0240
      VYI=SQRT(VI*VI-VXI*VXI)                                           INCE0250
      ITRY=0                                                            INCE0260
  111 VY=VYT-VYI                                                        INCE0270
      IF(VY.NE.0.) DELTI=-DYO/VY                                        INCE0280
      IF(DELTI.GE.0.) RETURN                                            INCE0290
      IF(ITRY.GT.0) RETURN                                              INCE0300
      ITRY=1                                                            INCE0310
      VYI=-VYI                                                          INCE0320
      GO TO 111                                                         INCE0330
  150 E=DYO*VI                                                          INCE0340
      D=DYO*VXT-DXO*VYT                                                 INCE0350
      TWOA=2.*(DXO*DXO+DYO*DYO)                                         INCE0360
      B=2.*D*DXO                                                        INCE0370
      C=(D-E)*(D+E)                                                     INCE0380
      RADCAL=B*B-2.*TWOA*C                                              INCE0390
      IF(RADCAL) 10,151,155                                             INCE0400
  151 ITRY=1                                                            INCE0410
      VYI=-B/TWOA                                                       INCE0420
  152 VY=VYT-VYI                                                        INCE0430
      IF(VY.NE.0.) DELTI=-DYO/VY                                        INCE0440
      IF(DELTI.LT.0.) GO TO 156                                         INCE0450
      VXI=-DXO/DYO*VY+VXT                                               INCE0460
      RETURN                                                            INCE0470
  155 RADCAL=SQRT(RADCAL)                                               INCE0480
      VYI=(RADCAL-B)/TWOA                                               INCE0490
      ITRY=0                                                            INCE0500
      GO TO 152                                                         INCE0510
  156 IF(ITRY.GT.0) RETURN                                              INCE0520
      ITRY=1                                                            INCE0530
      VYI=-(RADCAL+B)/TWOA                                              INCE0540
      GO TO 152                                                         INCE0550
      END                                                               INCE0560
```

```
      SUBROUTINE RELATE(NBG,NBU,INTVLB,LSYSB,KWSEG,BGCV,BXYZ,BPX1,      RELA0010
     1      IBSTAT,                                                     RELA0020
     2      KRU,NRG,NRTU,INTVLR,INVRTU,NRU,LSYST,KTSEG,ITRU,RPX1,       RELA0030
     3      RGCV,KRUK,RXYZ,RTUXYZ,IRVEC,ITVEC,IRSTAT,                   RELA0040
     4      INFORM,LBRSYS,WDATA,     IDU1)                              RELA0050
      COMMON /FTIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP         RELA0060
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             RELA0070
      COMMON/FCONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,PDMIN,          RELA0080
     1      AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)        RELA0090
      DIMENSION INTVLB(1),LSYSB(1),KWSEG(1),INTVLR(1),INVRTU(1),        RELA0100
     1     NRU(1),LSYST(1),KTSEG(1),ITRU(1),INFORM(1),LBRSYS(1),        RELA0110
     2     BGCV(8,5,1),BXYZ(IDU1,7),BPX1(12,1),RGCV(8,5,1),RPX1(12,1),  RELA0120
     3     RXYZ(IDU1,7),KRUK(4,1),RTUXYZ(8,1),WDATA(24,1),NBU(1)        RELA0130
     4,      IRSTAT(1),IRSTAT(1)
      ASSIGN 490 TO ITSW3
      IF(IRVEC.GT.0) GO TO 107
      ASSIGN 200 TO IRSW1
      ASSIGN 206 TO IRSW2
      ASSIGN 410 TO IRSW3
      GO TO 108
  107 ASSIGN 201 TO IRSW1
      ASSIGN 212 TO IRSW2
      ASSIGN 3 TO IRSW3
  108 IF(NRTU.EQ.0) GO TO 111
      ASSIGN 4 TO ITSW3
  111 IF(ITVEC.GT.0) GO TO 112
      ASSIGN 500 TO ITSW1
      ASSIGN 400 TO ITSW2
      GO TO 115
  112 ASSIGN 201 TO ITSW1
      ASSIGN 402 TO ITSW2
  115 CONTINUE
      DO 500 IBG=1,NBG
      K=INTVLP(IBG)
      IF(K.LT.0) GO TO 202
      GO TO IRSW1,(200,201)
  200 GO TO ITSW1,(201,500)
  201 ASSIGN 208 TO IBSW1
      ASSIGN 218 TO IBSW2
      ASSIGN 390 TO IBSW3
      ASSIGN 490 TO IBSW4
      ASSIGN 480 TO IBSW5
      GO TO 204
  202 ASSIGN 212 TO IBSW1
      ASSIGN 3 TO IBSW2
      ASSIGN 300 TO IBSW3
      ASSIGN 402 TO IBSW4
      ASSIGN 412 TO IBSW5
  204 LFG=IABS(K)
      K=NBU(IBG)
      IBA=K/100
      IBB=IBA+MOD(K,100)-1
      VXB=BGCV(LFG,1,IBG)
      VYB=BGCV(LFG,2,IBG)
      DO 490 IBU=IBA,IBB
      IF(IBSTAT(IBU).EQ.0) GO TO 490
      IS=LSYSB(IBU)
      LS=MOD(IS/100,10000)
      IF(LS.EQ.0) GO TO 490
      MS=LBRSYS(LS)
      MASK=MOD(MS/10, 1000)
      IF(MASK.EQ.0) GO TO 490
      K=LBRSYS(LS-1)
      RAIR=K/100000
      RSHIP=MOD(K/100,1000)
      GO TO IRSW2,(206,212)
  206 GO TO IRSW1,(208,212)
  208 IF(MOD(MASK/10,10) .EQ. 0) GO TO 490
  212 KB=MOD(MS,10)                                                     RELA0690
      XB=BXYZ(IBU,2)                                                    RELA0700
      YB=BXYZ(IBU,3)                                                    RELA0710
      ZB=BXYZ(IBU,4)                                                    RELA0720
      LPLTW=MOD(IS,100)                                                 RELA0730
      LS=LS-1                                                           RELA0740
      IS=IS/1000000                                                     RELA0750
      IF(KB.NE.NSUB) GO TO 213                                          RELA0760
      ASSIGN 309 TO ISBSW1                                              RELA0770
      GO TO 214                                                         RELA0780
  213 ASSIGN 202 TO ISBSW1                                              RELA0790
      HBM=ZB                                                            RELA0800
      IF(KB.NE.NAIR) HBM=BPX1(3,LPLTW)/FPNM                             RELA0810
      BLOSM=SQRT(HBM*(6880.+HBM))                                       RELA0820
      IF(MOD(MS/100000, 10) .EQ. 0) GO TO 214                           RELA0830
      DO 2136 KS=IS,LS                                                  RELA0840
      JS=LBRSYS(KS)                                                     RELA0850
      IF(MOD(JS,10) .NE. 1) GO TO 2136                                  RELA0860
      BETA=MOD(JS/1000000,1000)                                         RELA0870
      ASSIGN 304 TO IRDSW1                                              RELA0880
      ASSIGN 414 TO IRDSW2                                              RELA0890
      GO TO 215                                                         RELA0900
 2136 CONTINUE                                                          RELA0910
  214 ASSIGN 306 TO IRDSW1                                              RELA0920
      ASSIGN 416 TO IRDSW2                                              RELA0930
  215 IF(MOD(MS/1000000,10).EQ. 0) GO TO 2152                           RELA0940
      DO 2150 KS=IS,LS                                                  RELA0950
      JS=LBRSYS(KS)                                                     RELA0960
      IF(MOD(JS,10) .NE. 7) GO TO 2150                                  RELA0970
      SETA=MOD(JS/1000000,1000)                                         RELA0980
      SETA=SETA*.1                                                      RELA0990
      KSONR=1                                                           RELA1000
      GO TO 216                                                         RELA1010
 2150 CONTINUE                                                          RELA1020
 2152 SETA=0.                                                           RELA1030
      KSONR=0                                                           RELA1040
      IF(KB.EQ.NSUB) GO TO 490                                          RELA1050
  216 GO TO IBSW2,(218,3)                                               RELA1060
  218 GO TO IRSW3,(410,3)                                               RELA1070
    3 DO 390 IRG=1,NRG                                                  RELA1080
      LEGT=INTVLR(IRG)                                                  RELA1090
      IF(LEGT.LT.0) GO TO 300                                           RELA1100
      GO TO IBSW3,(390,300)                                             RELA1110
  300 K=NRU(IRG)                                                        RELA1120
      IRA=K/100                                                         RELA1130
      IRB=IRA+MOD(K,100)-1                                              RELA1140
      LEG=IABS(LEGT)                                                    RELA1150
      VXR=RGCV(LEG,1,IRG)                                               RELA1160
      VYR=RGCV(LEG,2,IRG)                                               RELA1170
      DO 380 IRU=IRA,IRB                                                RELA1180
      IF(IRSTAT(IRU).EQ.0) GO TO 380                                    RELA1190
      KR=MOD(KRUK(2,IRU)/1000000,10)                                    RELA1200
      IF(MOD(MASK/((10**KR)/10),10) .EQ. 0) GO TO 380                   RELA1210
      LPLTR=MOD(LSYST(IRU),100)                                         RELA1220
      XR=RXYZ(IRU,2)                                                    RELA1230
      YR=RXYZ(IRU,3)                                                    RELA1240
      ZR=RXYZ(IRU,4)                                                    RELA1250
      IF(KR.EQ.NSUB) GO TO 309                                          RELA1260
      GO TO ISBSW1,(302,309)                                            RELA1270
  302 HR=ZR                                                             RELA1280
      IF(HR.LT.HMIN) HR=HMIN                                            RELA1290
      RLOS=SQRT(HR*(6880.+HR))                                          RELA1300
      TOTLOS=RLOS+BLOSM                                                 RELA1310
      RADIUS=RSHIP                                                      RELA1320
      IF(KR.EQ.2) RADIUS=RAIR                                           RELA1330
C                                                                       RELA1340
      GO TO IRDSW1,(306,304)                                            RELA1350
  304 K=6- KR/1 + 1/KR                                                  RELA1360
      RCS=RPX1(K,LPLTR)                                                 RELA1370
      RADSEE=BETA*RCS**.25                                              RELA1380
      X=1.1547*TOTLOS                                                   RELA1390
      IF(RADSEE.GT.X) RADSEE=X                                          RELA1400
      GO TO 308                                                         RELA1410
  306 RADSEE=0.                                                         RELA1420
  308 SIZE=.5*(RPX1(1,LPLTR)+RPX1(2,LPLTR))                             RELA1430
      VISUAL= .165*SIZE                                                 RELA1440
      IF(VISUAL.GT.TOTLOS) VISUAL=TOTLOS                                RELA1450
      SIGHT=RADSEE                                                      RELA1460
      IF(SIGHT.LT.VISUAL) SIGHT=VISUAL                                  RELA1470
      IF(SIGHT.GT.RADIUS) SIGHT=RADIUS                                  RELA1480
      GO TO 311                                                         RELA1490
  309 SIGHT=SETA                                                        RELA1500
  311 DXO=XR-XB                                                         RELA1510
      DYO=YR-YB                                                         RELA1520
      VX=VXR-VXB                                                        RELA1530
      VY=VYR-VYB                                                        RELA1540
      CALL NQUIRE(INFORM,KWSEG,IBU,IRU,ISEG)                            RELA1550
      CALL GRTEST(DXO,DYO,VX,VY,SIGHT,TINT,TSTART,TSTOP)                RELA1560
      IF(ABS(TINT) .LT. .003) GO TO 312                                 RELA1570
      IF(TSTART.GT.TIMEB) GO TO 312                                     RELA1580
      IF(ISEG.GT.0) GO TO 316                                           RELA1590
      CALL ADDSEG(INFORM,KWSEG,KTSEG,IBU,IRU,ISEG)                      RELA1600
  312 IF(ISEG.EQ.0) GO TO 380                                           RELA1610
  316 CALL UPSEG(INFORM,ISEG,TSTART,TSTOP)                              RELA1620
  380 CONTINUE                                                          RELA1630
  390 CONTINUE                                                          RELA1640
      GO TO ITSW3,(4,490)                                               RELA1650
    4 GO TO ITSW2,(402,400)                                             RELA1660
  400 GO TO IBSW4,(402,490)                                             RELA1670
  402 IF(MOD(MASK/10,10) .EQ. 0) GO TO 490                              RELA1680
  410 DO 480 IRU=1,NRTL                                                 RELA1690
      LEGT=INVRTU(IRU)                                                  RELA1700
      IF(LEGT) 412,480,411                                              RELA1710
  411 GO TO IBSW5,(480,412)                                             RELA1720
  412 XR=RTUXYZ(1,IRU)                                                  RELA1730
      YR=RTUXYZ(2,IRU)                                                  RELA1740
      ZR=RTUXYZ(3,IRU)                                                  RELA1750
      IF(ZR.LT.0.) GO TO 480                                            RELA1760
      VXR=RTUXYZ(4,IRU)                                                 RELA1770
      VYR=RTUXYZ(5,IRU)                                                 RELA1780
      RLOS=SQRT(ZR*(6880.+ZR))                                          RELA1790
      TOTLOS=RLOS+BLOSM                                                 RELA1800
C                                                                       RELA1810
      DXO=XR-XB                                                         RELA1820
      DYO=YR-YB                                                         RELA1830
      VX=VXR-VXB                                                        RELA1840
      VY=VYR-VYB                                                        RELA1850
      ITU=ITRU(IRU)                                                     RELA1860
      KTU=MOD(ITU,10)                                                   RELA1870
      LPLTR=MOD(ITU/1000,100)                                           RELA1880
      NTRAJ=MOD(ITU/10,100)                                             RELA1890
      GO TO IRDSW2,(414,416)                                            RELA1900
  414 IF(KTU.EQ.2) GO TO 415                                            RELA1910
      RCS=WDATA(19,NTRAJ)                                               RELA1920
      GO TO 4150                                                        RELA1930
  415 RCS=RPX1(4,LPLTR)                                                 RELA1940
 4150 RADSEE=BETA*RCS**.25                                              RELA1950
      X=1.1547*TOTLOS                                                   RELA1960
      IF(RADSEE.GT.X) RADSEE=X                                          RELA1970
      GO TO 418                                                         RELA1980
  416 RADSEE=0.                                                         RELA1990
  418 IF(KTU.EQ.2) GO TO 419                                            RELA2000
      SIZE= .5*(WDATA(16,NTRAJ)+WDATA(17,NTRAJ))                        RELA2010
      GO TO 4190                                                        RELA2020
  419 SIZE= .5*(RPX1(1,LPLTR)+RPX1(2,LPLTR))                            RELA2030
 4190 VISUAL= .165*SIZE                                                 RELA2040
      IF(VISUAL.GT.TOTLOS) VISUAL=TOTLOS                                RELA2050
      SIGHT=RADSEE                                                      RELA2060
      IF(SIGHT.LT.VISUAL) SIGHT=VISUAL                                  RELA2070
      IF(SIGHT.GT.RAIR) SIGHT=RAIR                                      RELA2080
      JRU=IRU+KRU                                                       RELA2090
      CALL NQUIRE(INFORM,KWSEG,IBU,JRU,ISEG)                            RELA2100
      CALL GPTEST(DXO,DYO,VX,VY,SIGHT,TINT,TSTART,TSTOP)                RELA2110
      IF(ABS(TINT) .LT. .003) GO TO 420                                 RELA2120
      IF(TSTART.GT.TIMEB) GO TO 420                                     RELA2130
      IF(ISEG.GT.0) GO TO 424                                           RELA2140
      CALL ADDSEG(INFORM,KWSEG,KTSEG,IBU,JRU,ISEG)                      RELA2150
  420 IF(ISEG.EQ.0) GO TO 480                                           RELA2160
  424 CALL UPSEG(INFORM,ISEG,TSTART,TSTOP)                              RELA2170
  480 CONTINUE                                                          RELA2180
  490 CONTINUE                                                          RELA2190
  500 CONTINUE                                                          RELA2200
      RETURN                                                            RELA2210
      END                                                               RELA2220


      SUBROUTINE NQUIRE(INFORM,ISNSEG,ISENS,ITGT,ISEG)                  NQUI0010
      COMMON/STRCON/ JPK,IPK                                            NQUI0020
      DIMENSION INFORM(1),ISNSEG(1)                                     NQUI0030
      IPOINT=ISNSEG(ISENS)                                              NQUI0040
      ISEG=IPOINT/IPK                                                   NQUI0050
    1 IF(ISEG.EQ.0) RETURN                                              NQUI0060
      JTGT=MOD(INFORM(ISEG),IPK)                                        NQUI0070
      IF(JTGT.EQ.ITGT) RETURN                                           NQUI0080
      JPOINT=INFORM(ISEG+1)                                             NQUI0090
      ISEG=MOD(JPOINT,IPK)                                              NQUI0100
      GO TO 1                                                           NQUI0110
      END                                                               NQUI0120


      SUBROUTINE ADDSEG(INFORM,         ISNSEG,ITGSEG,ISENS,ITGT,ISEG)  ADDS0010
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             ADDS0020
      COMMON/STRCON/  JPK,IPK                                           ADDS0030
      COMMON/INFO/  LENIFO,LENSEG,MAXSEG,NEXSEG,LSTSEG,LZDSEG           ADDS0040
      DIMENSION INFORM(1),            ISNSEG(1),ITGSEG(1)               ADDS0050
      DATA IPR/0/                                                       ADDS0060
      IF(NEXSEG.EQ.0) GO TO 180                                         ADDS0070
      ISEG=NEXSEG                                                       ADDS0080
      NEXSEG=INFORM(ISEG)                                               ADDS0090
      GO TO 190                                                         ADDS0100
  180 LSEG=LSTSEG+LENSEG                                                ADDS0110
      IF(LSEG.GT.MAXSEG) GO TO 210                                      ADDS0120
      ISEG=LSEG                                                         ADDS0130
      LSTSEG=LSEG                                                       ADDS0140
  190 INFORM(ISEG)=ISENS*IPK+ITGT                                       ADDS0150
      DO 200 IADD=4,LENSEG                                              ADDS0160
  200 INFORM(ISEG+IADD-1)=0                                             ADDS0170
      IPOINT=ISNSEG(ISENS)                                              ADDS0180
      IADD=1                                                            ADDS0190
      IST=ISENS                                                         ADDS0200
  201 IF(IPOINT.GT.0) GO TO 203                                         ADDS0210
      JPOINT=ISEG*IPK+ISEG                                              ADDS0220
      INFORM(ISEG+IADD)=0                                               ADDS0230
      IF(IADD.EQ.2) GO TO 207                                           ADDS0240
      GO TO 205                                                         ADDS0250
  203 JPOINT=IPOINT/IPK*IPK+ISEG                                        ADDS0260
      LSEG=MOD(IPOINT,IPK)                                              ADDS0270
      INFORM(ISEG+IADD)=LSEG*IPK                                        ADDS0280
      INFORM(LSEG+IADD)=INFORM(LSEG+IADD)+ISEG                          ADDS0290
      IF(IADD.EQ.2) GO TO 207                                           ADDS0300
  205 ISNSEG(ISENS)=JPOINT                                              ADDS0310
      IPOINT=ITGSEG(ITGT)                                               ADDS0320
      IADD=2                                                            ADDS0330
      IST=ITGT                                                          ADDS0340
      GO TO 201                                                         ADDS0350
  207 ITGSEG(ITGT)=JPOINT                                               ADDS0360
      RETURN                                                            ADDS0370
  210 ISEG=0                                                            ADDS0380
      IF(IPR.GT.0) RETURN                                               ADDS0390
      IPR=1                                                             ADDS0400
      CALL PAGE                                                         ADDS0410
      WRITE(N6,6000)                                                    ADDS0420
 6000 FORMAT(////'                      FULL INFORMATION ARRAY')        ADDS0430
      RETURN                                                            ADDS0440
      END                                                               ADDS0450
      SUBROUTINE UPSEG (INFORM,ISEG,TSTART,TSTOP)                        UPSE0010
      DIMENSION  INFORM(1)                                               UPSE0020
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP          UPSE0030
      ITA=(TSTART-TBEGIN)*1000.                                          UPSE0040
      ITB=(TSTOP-TBEGIN)*1000.                                           UPSE0050
      INFORM(ISEG+4)=ITA*100000+ITB                                      UPSE0060
      INFORM(ISEG+5)=0                                                   UPSE0070
      RETURN                                                             UPSE0080
      END                                                                UPSE0090


      SUBROUTINE DETECT(NWU,NWTU,LSYSW,KWSEG,WRAD,WXYZ,KWUK,             DETE0010
     1     INTVLW,WGCV,WPX1,ENV,MAXWTU,IWSTAT,TUWXYZ,KSYSW,IWTU,         DETE0020
     2    NTU,NTTU,LSYST,KTSEG,TXYZ,KTUK,INTVLT,TGCV,TPX1,AJAM,          DETE0030
     3     ITTU,TUTXYZ,ITSTAT,KHVT,                                      DETE0040
     4     TDATA,LBRSYS,INFORM,CMRAD,                       IDU1)        DETE0050
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              DETE0060
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP          DETE0070
      COMMON/ECONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,PDMIN,           DETE0080
     1     AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)          DETE0090
      COMMON/STRCON/ JPK,IPK                                             DETE0100
      COMMON /WORK/    IWORK(100),WORKI(100),WORKJ(100),JWORK(100)       DETE0110
      DIMENSION KWORK(500)                                               DETE0120
      EQUIVALENCE (IWORK(1),KWORK(1))                                    DETE0130
      DIMENSION KHVT(1)                                                  DETE0140
      DIMENSION LSYSW(1),LBRSYS(1),KWSEG(1),WRAD(36,1),INFORM(1),        DETE0150
     1     WXYZ(IDU1,7),TXYZ(IDU1,7),KWUK(4,1),INTVLW(1),                DETE0160
     2     WGCV(8,5,1),WPX1(12,1),LSYST(1),KTSEG(1),                     DETE0170
     3     KTUK(4,1),INTVLT(1),TGCV(8,5,1),TPX1(12,1),                   DETE0180
     4     ENV(1),AJAM(6,3,1),ITTU(1),TUTXYZ(8,1),TDATA(24,1)            DETE0190
      DIMENSION TAR(6),ETC(4),TRAD(10),RANGE(10),PDTECT(10),ANG(10),     DETE0200
     1     DYN(10),ITSTAT(1),IWSTAT(1),KSYSW(1),TUWXYZ(8,1),             DETE0210
     2     CMRAD(36,1),IWTU(1),RADCOL(36)                                DETE0220
      NAMELIST/RADR/ IW,KW,WZ,IRAD,ITGT,KT,TZ,RCS,ALTGT,RDRNGE,          DETE0230
     1     GR,PDRAD,BETA,ALFA,FREQ,BEAM,IJAM,KJAM,JAMON,JAM,JAMR,        DETE0240
     2     ETC,TAR,TIME                                                  DETE0250
      NAMELIST/DTECT/ TIME,IW,ITGT,KPD,PD,GR                             DETE0260
C                                                                        DETE0270
      ICLASS=IMISC(21)                                                   DETE0280
      IPRSAV=IPRAD                                                       DETE0290
      ASSIGN 170 TO LPRAD                                                DETE0300
      IF(IMISC(3).GT.1) ASSIGN 1700 TO LPRAD                             DETE0310
      CALL ERRSET(208,300,-1,1,0,0)                                      DETE0320
      TRAD(1)=TIME                                                       DETE0330
      TAR(5)=0.                                                          DETE0340
      TAR(6)=0.                                                          DETE0350
      ITIME=(TIME-TBEGIN)*1000.                                          DETE0360
      ICTIME=ITIME+IMISC(1)                                              DETE0370
      JAMON=IMISC(2)                                                     DETE0380
      ASSIGN 105 TO LJSET                                                DETE0390
      ASSIGN 171 TO LWSET                                                DETE0400
      IW=1                                                               DETE0410
   10 IPSEG=KWSEG(IW)                                                    DETE0420
      IF(IPSEG.LE.0) GO TO 200                                           DETE0430
      IS=LSYSW(IW)                                                      DETE0440
      LPLTW=MOD(IS,100)                                                 DETE0450
      IS=IS/100                                                         DETE0460
      IF(IS.EQ.0) GO TO 200                                             DETE0470
      ASSIGN 14 TO IWSET                                                DETE0480
      ASSIGN 120 TO JWSET                                               DETE0490
      ASSIGN 160 TO KWSET                                               DETE0500
      ISEG=IPSEG/IPK                                                    DETE0510
    1 IF(ISEG.EQ.0) GO TO 200                                           DETE0520
      ITGT=MOD(INFORM(ISEG),IPK)                                        DETE0530
      ISTAT=ITSTAT(ITGT)                                                DETE0540
      IF(ISTAT.LE.0) GO TO 180                                          DETE0550
      LOSTA=INFORM(ISEG+4)                                              DETE0560
      LOSTB=MOD(LOSTA,100000)                                           DETE0570
      LOSTA=LOSTA/100000                                                DETE0580
      IF(ITIME.LT.LOSTA) GO TO 18                                       DETE0590
      IF(ITIME.GT.LOSTB) GO TO 180                                      DETE0600
      IDTECT=INFORM(ISEG+5)                                             DETE0610
      IF(IDTECT) 11,12,11                                               DETE0620
   11 NEXT=IABS(IDTECT)/10000                                           DETE0630
      IF(ITIME.LT.NEXT) GO TO 18                                        DETE0640
   12 GO TO JWSET,(120,121)                                             DETE0650
  120 ASSIGN 121 TO JWSET                                               DETE0660
      WX=WXYZ(IW,2)                                                     DETE0670
      WY=WXYZ(IW,3)                                                     DETE0680
      WZ=WXYZ(IW,4)                                                     DETE0690
      IWG=KWUK(3,IW)                                                    DETE0700
      LEGW=IABS(INTVLW(IWG))                                            DETE0710
      WVX=WGCV(LEGW,1,IWG)                                              DETE0720
      WVY=WGCV(LEGW,2,IWG)                                              DETE0730
      ISB=MOD(IS,10000)                                                 DETE0740
      MASK=LBRSYS(ISB)                                                  DETE0750
      KW=MOD(MASK,10)                                                   DETE0760
      MASK=MASK/10                                                      DETE0770
      ISA=IS/10000                                                      DETE0780
      KRADR=MOD(MASK/10000,10)                                          DETE0790
      KSONR=MOD(MASK/100000,10)                                         DETE0800
      ISB=ISB-1                                                         DETE0810
      IF(KSONR.EQ.0) GO TO 1203                                         DETE0820
      DO 1202 IS=ISA,ISB                                                DETE0830
      JSYS=LBRSYS(IS)                                                   DETE0840
      IF(MOD(JSYS,10).EQ.7) GO TO 1204                                  DETE0850
 1202 CONTINUE                                                          DETE0860
      KSONR=0                                                           DETE0870
 1203 SETA=0.                                                           DETE0880
      GO TO 1205                                                        DETE0890
 1204 ISON=MOD(JSYS/10000,100)                                          DETE0900
      SETA=MOD(JSYS/1000000,1000)                                       DETE0910
      SETA=SETA*.1                                                      DETE0920
      IF(SETA.EQ.0.) KSONR=0                                            DETE0930
 1205 IF(KRADR.EQ.0) GO TO 1207                                         DETE0940
      DO 1206 IS=ISA,ISB                                                DETE0950
      JSYS=LBRSYS(IS)                                                   DETE0960
      IF(MOD(JSYS,10).EQ.1) GO TO 1208                                  DETE0970
 1206 CONTINUE                                                          DETE0980
```

```
      KRADR=0                                                           DETE0990
 1207 ASSIGN 170 TO IWSET                                               DETE1000
      GO TO 121                                                         DETE1010
 1208 HMAST=0.                                                          DETE1020
      IF(KW.NE.NAIR) HMAST=WPX1(3,LPLTW)                                DETE1030
      IRAD=MOD(JSYS/10000,100)                                          DETE1040
      DO 1209 I=1,36                                                    DETE1050
 1209 RADCOL(I)=WRAD(I,IRAD)                                            DETE1060
 1210 BETA=RADCOL(1)                                                    DETE1070
      ALFA=RADCOL(2)                                                    DETE1080
      FREQ=30000./RADCOL(3)                                             DETE1090
      FTC(4)=RADCOL(36)                                                 DETE1100
      BEAM=.5*(RADCOL(22)/57.3)                                         DETE1110
      WZF    =WZ*FPNM+HMAST                                             DETE1120
      IF(WZF.LE.0.) ASSIGN 170 TO IWSET                                 DETE1130
      FTC(1)=WZF                                                        DETE1140
  121 IF(ITGT.LE.NTU)GO TO 130                                          DETE1150
      JTGT=ITGT-NTU                                                     DETE1160
      ITU=ITTU(JTGT)                                                    DETE1170
      KT=MOD(ITU,10)                                                    DETE1180
      TX=TUTXYZ(1,JTGT)                                                 DETE1190
      TY=TUTXYZ(2,JTGT)                                                 DETE1200
      TZ=TUTXYZ(3,JTGT)                                                 DETE1210
      IF(TZ.GT.0.) GO TO 124                                            DETE1220
      PD=0.                                                             DETE1230
      KPD=0                                                             DETE1240
      GO TO 174                                                         DETE1250
  124 TVX=TUTXYZ(4,JTGT)                                                DETE1260
      TVY=TUTXYZ(5,JTGT)                                                DETE1270
      IF(KT.EQ.NAIR) GO TO 125                                          DETE1280
      ICOL=MOD(ITU/10,100)                                              DETE1290
      RCS=TDATA(19,ICOL)                                                DETE1300
      ALTGT=.5*(TDATA(16,ICOL)+TDATA(17,ICOL))                          DETE1310
      GO TO 135                                                         DETE1320
  125 IPLTT=MOD(ITU/1000,100)                                           DETE1330
      GO TO 131                                                         DETE1340
  130 KT=MOD(KTUK(2,ITGT)/1000000,10)                                   DETE1350
      TX=TXYZ(ITGT,2)                                                   DETE1360
      TY=TXYZ(ITGT,3)                                                   DETE1370
      TZ=TXYZ(ITGT,4)                                                   DETE1380
      IF(TZ.LT.HMIN) TZ=HMIN                                            DETE1390
      IPLTT=MOD(LSYST(ITGT),100)                                        DETE1400
      ITG=KTUK(3,ITGT)                                                  DETE1410
      LEGT=IABS(INTVLT(ITG))                                            DETE1420
      TVX=TGCV(LEGT,1,ITG)                                              DETE1430
      TVY=TGCV(LEGT,2,ITG)                                              DETE1440
      IF(KT.EQ.NSUB) GO TO 135                                          DETE1450
  131 ALTGT=.5*(TPX1(1,IPLTT)+TPX1(2,IPLTT))                            DETE1460
      IF(KT.EQ.NAIR) GO TO 134                                          DETE1470
      RCS=TPX1(6,IPLTT)                                                 DETE1480
      GR=SQRT((TX-WX)**2+(TY-WY)**2)                                    DETE1490
      CALL SHIPXS(WZF,GR,RCS,TPX1(3,IPLTT),XCS,C,ICK)                   DETE1500
      IF(ICK.EQ.0) RCS=XCS                                              DETE1510
      GO TO 135                                                         DETE1520
  134 RCS=TPX1(4,IPLTT)                                                 DETE1530
  135 DX=TX-WX                                                          DETE1540
      DY=TY-WY                                                          DETE1550
      GR2=DX*DX+DY*DY                                                   DETE1560
      GR=SQRT(GR2)                                                      DETE1570
      IF(KW.NE.NSUB .AND. KT.NE.NSUB) GO TO 139                         DETE1580
      IF(KSONR.EQ.0) GO TO 180                                          DETE1590
      PDSON=0.                                                          DETE1600
      PD=GR2/(SETA*SETA)                                                DETE1610
      IF(PD.LT.100.) PDSON=EXP(-.693*PD)                                DETE1620
      KPD=20                                                            DETE1630
      PD=PDSON                                                          DETE1640
      GO TO 174                                                         DETE1650
  139 RV=.165*ALTCT                                                     DETE1660
      PDVIS=0.                                                          DETE1670
      PD=GR2/(RV*RV)                                                    DETE1680
      IF(PD.LT.100.) PDVIS=EXP(-.693*PD)                                DETE1690
      PDRAD=0.                                                          DETE1700
      GO TO IWSET,(14,145,170)                                          DETE1710
   14 ASSIGN 145 TO IWSET                                               DETE1720
      GO TO LJSET,(105,111)                                             DETE1730
  105 ASSIGN 111 TO LJSET                                               DETE1740
      ITYPE=5                                                           DETE1750
      JAM=0                                                             DETE1760
      DO 110 IT=1,NTU                                                   DETE1770
      IS=LSYST(IT)/100                                                  DETE1780
      IF(IS.EQ.0) GO TO 110                                             DETE1790
      IA=IS/10000                                                       DETE1800
      IB=MOD(IS,10000)                                                  DETE1810
      MASK=LBRSYS(IB)/10                                                DETE1820
      IF(MOD(MASK/1000,10) .EQ. 0) GO TO 110                            DETE1830
      IB=IB-1                                                           DETE1840
      DO 1000 IS=IA,IB                                                  DETE1850
      JSYS=LBRSYS(IS)                                                   DETE1860
      IF(MOD(JSYS,10) .NE. ITYPE) GO TO 1000                            DETE1870
      IF(JSYS/1000000000+JAMON.EQ.0) GO TO 1000                         DETE1880
      ICOL=MOD(JSYS/10000,100)                                          DETE1890
      JAM=JAM+1                                                         DETE1900
      JWORK(JAM)=IT*100+ICOL                                            DETE1910
 1000 CONTINUE                                                          DETE1920
  110 CONTINUE                                                          DETE1930
  111 CONTINUE                                                          DETE1940
      IF(JAM.EQ.0) GO TO 145                                            DETE1950
      JAMR=0                                                            DETE1960
      DO 140 JM=1,JAM                                                   DETE1970
      IJ=JWORK(JM)                                                      DETE1980
      IT=IJ/100                                                         DETE1990
      IJ=MOD(IJ,100)                                                    DETE2000
      DO 141 K=1,3                                                      DETE2010
      IF(FREQ.LT.AJAM(5,K,IJ)) GO TO 141                                DETE2020
      IF(FREQ.GT.AJAM(6,K,IJ)) GO TO 141                                DETE2030
      KJ=K                                                              DETE2040
      GO TO 142                                                         DETE2050
  141 CONTINUE                                                          DETE2060
      GO TO 140                                                         DETE2070
  142 XJ=TXYZ(IT,2)                                                     DETE2080
```

```
      YJ=TXYZ(IT,3)                                                     DETE2090
      DXJ=XJ-WX                                                         DETE2100
      DYJ=YJ-WY                                                         DETE2110
C                                                                       DETE2120
C                                                                       DETE2130
C                                                                       DETE2140
      ANGJ=ATAN2(DXJ,DYJ)                                               DETE2150
      SOJR=SQRT(DXJ*DXJ+DYJ*DYJ)                                        DETE2160
      JAMR=JAMR+1                                                       DETE2170
      IWORK(JAMR)=KJ*100+IJ                                             DETE2180
      WORKI(JAMR)=ANGJ                                                  DETE2190
      WORKJ(JAMR)=SOJR                                                  DETE2200
  140 CONTINUE                                                          DETE2210
      IF(JAMR.GT.0) ASSIGN 16 TO KWSET                                  DETE2220
  145 GO TO KWSET,(16,160)                                              DETE2230
   16 ANGT=ATAN2(DX,DY)                                                 DETE2240
      DO 161 IJ=1,JAMR                                                  DETE2250
      A=ABS(ANGT-WORKI(IJ))                                             DETE2260
      IF(A.GT.PI) A=TWOPI-A                                             DETE2270
      IF(A.GT.BEAM) GO TO 161                                           DETE2280
      IJAM=IWORK(IJ)                                                    DETE2290
      ETC(2)=WORKJ(IJ)                                                  DETE2300
      GO TO 162                                                         DETE2310
  161 CONTINUE                                                          DETE2320
  160 RDRNGE=BETA*RCS**.25                                              DETE2330
      ETC(2)=0.                                                         DETE2340
      ETC(3)=0.                                                         DETE2350
      IJAM=1                                                            DETE2360
      KJAM=1                                                            DETE2370
      GO TO 165                                                         DETE2380
  162 KJAM=IJAM/100                                                     DETE2390
      IJAM=MOD(IJAM,100)                                                DETE2400
      ETC(3)=2.                                                         DETE2410
      GJ2=10.**(.1*AJAM(2,KJAM,IJAM))                                   DETE2420
      PJ2=AJAM(1,KJAM,IJAM)                                             DETE2430
      RDRNGE=ALFA*(500./PJ2*(1.26/GJ2)**.25)                            DETE2440
  165 IF(RDRNGE.LT.GR) GO TO 170                                        DETE2450
      IF(IPRAD.GE.0) GO TO 17                                           DETE2460
      PDRAD=EXP(-.693*GR2/(RDRNGE*RDRNGE))                              DETE2470
      GO TO 170                                                         DETE2480
   17 VX=TVX-WVX                                                        DETE2490
      VY=TVY-WVY                                                        DETE2500
      TZF=TZ*FPNM                                                       DETE2510
      DGRA=VX*DX+VY*DY                                                  DETE2520
      VC=0.                                                             DETE2530
      IF(GR.GT.0.) VC=DGRA/GR                                           DETE2540
      DZ=ABS(WZF-TZF)/FPNM                                              DETE2550
      RANGE(1)=SQRT(GR2+DZ*DZ)                                          DETE2560
      TAR(1)=ALTGT                                                      DETE2570
      TAR(2)=RCS                                                        DETE2580
      TAR(3)=TZF                                                        DETE2590
      TAR(4)=VC                                                         DETE2600
      CALL RADAR(RADCOL,ENV,ETC,TAR,AJAM(1,KJAM,IJAM),TRAD,RANGE,       DETE2610
     1     PDTECT,ANG,DYN,ICK,ISCAN,IPRAD)                              DETE2620
      PDRAD=PDTECT(ISCAN+1)                                             DETE2630
      GO TO LPRAD,(1700,170)                                            DETE2640
 1700 WRITE(N6,RADR)                                                    DETE2650
  170 GO TO LWSET,(171,230)                                             DETE2660
  171 IF(PDVIS.GT.PDRAD) GO TO 172                                      DETE2670
      PD=PDRAD                                                          DETE2680
      KPD=10                                                            DETE2690
      GO TO 174                                                         DETE2700
  172 PD=PDVIS                                                          DETE2710
      KPD=0                                                             DETE2720
  174 IPD=100.*PD                                                       DETE2730
      IF(IPD.GT.99) IPD=99                                              DETE2740
      IDTECT=10000*IDTIME+100*KPD+IPD                                   DETE2750
      IF(PD.GE.PDMIN) GO TO 175                                         DETE2760
      IDTECT=-IDTECT                                                    DETE2770
      GO TO 176                                                         DETE2780
  175 IF(KT.NE.1) GO TO 1755                                            DETE2790
      IF(IPD.LT.ICLASS) GO TO 1755                                      DETE2800
      JPD=KHVT(ITGT)                                                    DETE2810
      IF(JPD/100 .EQ. 0) KHVT(ITGT)=JPD+100                             DETE2820
 1755 IF(ISTAT.GT.1) GO TO 176                                          DETE2830
C                                                                       DETE2840
C  DETECTION GAINED                                                     DETE2850
C                                                                       DETE2860
      ITSTAT(ITGT)=2                                                    DETE2870
      IF(IMISC(3).GT.0) WRITE(N6,DTECT)                                 DETE2880
  176 INFORM(ISEG+5)=IDTECT                                             DETE2890
   18 ISEG=MOD(INFORM(ISEG+1),IPK)                                      DETE2900
      GO TO 1                                                           DETE2910
  180 INFORM(ISEG+4)=2147400000                                         DETE2920
      INFORM(ISEG+5)=0                                                  DETE2930
      GO TO 18                                                          DETE2940
  200 IW=IW+1                                                           DETE2950
      IF(IW.LE.NWU) GO TO 10                                            DETE2960
      IF(NWTU.EQ.0) GO TO 240                                           DETE2970
      IF(IMISC(6).LT.0) GO TO 240                                       DETE2980
      ASSIGN 230 TO LWSET                                               DETE2990
      FMAST=0.                                                          DETE3000
      IW=0                                                              DETE3010
      IF(IPRAD.LT.0) IPRAD=0                                            DETE3020
  220 IW=IW+1                                                           DETE3030
      IF(IW.GT.NWTU) GO TO 240                                          DETE3040
      IF(IWSTAT(IW+NWU).LE.0) GO TO 220                                 DETE3050
      ITU=IWTU(IW)                                                      DETE3060
      IF(MOD(ITU,10).NE.4) GO TO 220                                    DETE3070
      IF(MOD(IWTU(MAXWTU+IW)/100000000,10).NE.1) GO TO 220              DETE3080
      IROW=MOD(ITU/1000,100)                                            DETE3090
      IRAD=MOD(KSYSW(IROW)/100,100)                                     DETE3100
      IF(IRAD.EQ.0) GO TO 228                                           DETE3110
      ITGT=ITU/100000                                                   DETE3120
      KW=MOD(ITU,10)                                                    DETE3130
      WX=TUWXYZ(1,IW)                                                   DETE3140
      WY=TUWXYZ(2,IW)                                                   DETE3150
      WZ=TUWXYZ(3,IW)                                                   DETE3160
      WVX=TUWXYZ(4,IW)                                                  DETE3170
      WVY=TUWXYZ(5,IW)                                                  DETE3180
      ASSIGN 14 TO IWSET                                                DETE3190
      ASSIGN 160 TO KWSET                                               DETE3200
      DO 225 I=1,36                                                     DETE3210
  225 RADCOL(I)=CMRAD(I,IRAD)                                           DETE3220
      GO TO 1210                                                        DETE3230
  228 IPD=99                                                            DETE3240
      GO TO 235                                                         DETE3250
  230 IJAM=ETC(3)+.5                                                    DETE3260
      IPD=PDRAD*100.                                                    DETE3270
      IF(IPD.GT.99) IPD=99                                              DETE3280
      IPD=IJAM*100+IPD                                                  DETE3290
      IF(IMISC(3).GT.0) WRITE(N6,RADR)                                  DETE3300
  235 ITU=IWTU(MAXWTU+IW)                                               DETE3310
      IWTU(MAXWTU+IW)=IPD*1000000+MOD(ITU,1000000)                      DETE3320
      GO TO 220                                                         DETE3330
  240 CALL FPRSET(208,300,300,2,0,0)                                    DETE3340
      IPRAD=IPRSAV                                                      DETE3350
      RETURN                                                            DETE3360
      END                                                               DETE3370


       SUBROUTINE  SHIPXS(H1,R,SIG,  HS,SIGA,H,IXX)                     SHXS0010
C  PGM=NXX. I.O.G.        VER.1  7-16-73       FORTRAN IV        EPCD   SHXS0020
C  TO FIND THE VISIBLE RADAR X-SECT. OF A SURFACE SHIP                  SHXS0030
C  FOR                                                                  SHXS0040
C      H1   = RADAR HEIGHT, FT.            / R     = RANGE TO SHIP,NM   SHXS0050
C      SIG  = RADAR X-SECT.OF SHIP, M2 / C     = AREA DIST.CONSTANT, WHERESHXS0060
C      HS   = VERTICAL DIM.SHIP,FT.    /         (1.0.LE.C.LE.2.0)      SHXS0070
C  OUTPUT                                                               SHXS0080
C      SIGA = RADAR X-SECTION VISIBLE ABOVE HORIZON, SQ.METERS          SHXS0090
C      H    = ALTITUDE OF CENTROID OF VISIBLE AREA, FT.ABOVE SURFACE    SHXS0100
C      IXX  = +1  IF RANGE R.GT.RADAR HORIZON                           SHXS0110
C           = 0   IF O.K.                                               SHXS0120
C***                                                                    SHXS0130
       DATA  C/2.0/                                                     SHXS0140
 1     IF (HS.GT.0.)  GO TO 8                                           SHXS0150
       IXX = -1                                                         SHXS0160
       RETURN                                                           SHXS0170
C** RADAR HORIZON                                                        SHXS0180
 8     IF (H1.LE.0.)  H1 = 0.                                           SHXS0190
       RH1  = 1.2289*SQRT(H1)                                           SHXS0200
       HB   = 0.7 * HS                                                  SHXS0210
       RH   = RH1 + 1.2289*SQRT(HB)                                     SHXS0220
       IF (R.LT.RH)  GO TO 10                                           SHXS0230
C*** NOT VISIBLE                                                         SHXS0240
       IXX = +1                                                         SHXS0250
       RETURN                                                           SHXS0260
C*** COMPLETELY VISIBLE                                                  SHXS0270
 10    IXX  = 0                                                         SHXS0280
       IF (R.GT.RH1+.0001)  GO TO 12                                    SHXS0290
       SIGA = SIG                                                       SHXS0300
       H    = HB / (C+1.0)                                              SHXS0310
       RETURN                                                           SHXS0320
C** PARTIALLY VISIBLE                                                    SHXS0330
```

```
12     R2    = R - RH1                                                   SHXS0340
       HH    = (R2 / 1.2289)**2                                          SHXS0350
       IF (HH.GT.HB)  HH = HB                                            SHXS0360
       HA    = HB - HH                                                   SHXS0370
       SIGA = SIG*( (HA/HB)**C)                                          SHXS0380
       H     = HH + HA / (C+1.0)                                         SHXS0390
       RETURN                                                            SHXS0400
       END                                                               SHXS0410


      SUBROUTINE ALLOCA(NEXTU,LASTU,MAXTU,IWSTAT,LSYSW,IWTU,INVWTU,      ALL00010
     1  TUWXYZ,WPX1,WXYZ,WRAD,ENV,KWSEG,KWU,KWUK,NWUG,                   ALL00020
     2  NTU,NTTU,KTSEG,ITSTAT,INTVLT,TUTXYZ,KTUK,TGCV,TXYZ,KHVT,         ALL00030
     3      AJAM,ITTU,TPX1,LSYST,INVTTU,                                 ALL00040
     4      TTRAJ,DATASM,WPNDAT,SAMDAT,LBRSYS,INFORM,IDU1)               ALL00050
      COMMON/ECONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,PDMIN,           ALL00060
     1      AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)         ALL00070
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              ALL00080
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP          ALL00090
      COMMON /STRCON/ JPK,IPK                                            ALL00100
      COMMON /WORK/ IWORK(50),JWORK(50),WORKI(50),WORKJ(50),             ALL00110
     1      LSHIP(40),LDET(220),LACVEC(40)                               ALL00120
      DIMENSION KWORK(500)                                               ALL00130
      EQUIVALENCE (IWORK(1),KWORK(1))                                    ALL00140
      DIMENSION IWSTAT(1),LSYSW(1),IWTU(1),INVWTU(1),TUWXYZ(8,1),        ALL00150
     1      TTRAJ(50,1),DATASM(24,1),WPNDAT(15,1),SAMDAT(20,1),          ALL00160
     2      WPX1(12,1),WXYZ(IDU1,7),KTSEG(1),ITSTAT(1),INTVLT(1),        ALL00170
     3      TUTXYZ(8,1),KTUK(4,1),TGCV(8,5,1),TXYZ(IDU1,7),AJAM(6,3,1),  ALL00180
     4      WRAD(36,1),ITTU(1),TPX1(12,1),LSYST(1),ENV(1)                ALL00190
     5,     LBRSYS(1),INFORM(1),INVTTU(1),KWSEG(1)                       ALL00200
     6,     KWUK(4,1),NWUG(1),KHVT(1)                                    ALL00210
      DIMENSION TRAD(10),PDTECT(10),ANG(10),DYN(10),RANGE(10),ETC(4),    ALL00220
     1      TAR(6)                                                       ALL00230
C                                                                        ALL00240
      DIMENSION MXGAGE(5),LGRP(18)                                       ALL00250
      EQUIVALENCE (IMISC(4),ISIDE)                                       ALL00260
      EQUIVALENCE (IMISC(20),LRATE)                                      ALL00270
      NAMELIST/CMLNCH/ TIME,IWPN,ITU,IWPNTU,KW,KSYS,ITGT,KT,GR,IPD       ALL00280
      NAMELIST/ACVECT/ TIME,IWPN,ITU,IWPNTU,KW,KSYS,ITGT,KT,GR,IPD       ALL00290
      NAMELIST/SSWPN/  TIME,IWPN,KW,KSYS,ITGT,KT,GR,TRUNIN,IPD           ALL00300
      NAMELIST/  SAM/ TIME,IWPN,KSYS,TA,TFA,GR,ITGT,KT,PDT               ALL00310
      NAMELIST/SAGUN/ TIME,IWPN,KSYS,TA,TFA,GR,ITGT,KT                   ALL00320
      NAMELIST/AAMDEF/ TIME,IWPN,KSYS,TFA,GR,ITGT,KT                     ALL00330
      NAMELIST /SAMLST/ TIME,IWPN,ITGT,NWSTAT,GR,X,RDRNGE,PDT            ALL00340
C                                                                        ALL00350
      DATA MAXX/2147400000/                                              ALL00360
C                                                                        ALL00370
      DO 110 I=1,5                                                       ALL00380
      IS=IMISC(I+30)                                                     ALL00390
      LS=MOD(IS,100)                                                     ALL00400
      IS=IS/100                                                          ALL00410
      IF(IS.EQ.0) IS=LS                                                  ALL00420
      IF(ISIDE.EQ.1) LS=IS                                               ALL00430
  110 MXGAGE(I)=LS                                                       ALL00440
```

```
      ASSIGN 546 TO IRPASS                                              ALL00450
      ASSIGN 465 TO IRFAIL                                              ALL00460
      IF(IMISC(19).LT. 20) IMISC(19)=20                                 ALL00470
      RSTMIN=IMISC(19)                                                  ALL00480
      RSTMIN=RSTMIN*.001                                                ALL00490
      NHVT=MOD(KHVT(19),100)                                            ALL00500
      TRAD(1)=TIME                                                      ALL00510
      TAR(5)=0.                                                         ALL00520
      TAR(6)=0.                                                         ALL00530
      ICHEK=IMISC(29)                                                   ALL00540
      TBACK=TIME-TSTEP                                                  ALL00550
      TZERO=TIME-TBEGIN                                                 ALL00560
      JAMON=IMISC(2)                                                    ALL00570
      NLPR=IMISC(3)                                                     ALL00580
      ITIME=TZERO*1000.                                                 ALL00590
      ASSIGN 311 TO KSWT1                                               ALL00600
      NVEC=0                                                            ALL00610
      KMASK=12                                                          ALL00620
      DO 125 I=1,2                                                      ALL00630
      DO 120 IW=1,KWU                                                   ALL00640
      LS=LSYSW(IW)                                                      ALL00650
      ISB=MOD(LS/100,10000)                                             ALL00660
      IF(ISB.EQ.0) GO TO 120                                            ALL00670
      MASK=LPRSYS(ISB)                                                  ALL00680
      IF(MASK.NE.KMASK) GO TO 120                                       ALL00690
      IF(IWSTAT(IW).LE.0) GO TO 120                                     ALL00700
      NVEC=NVEC+1                                                       ALL00710
      LACVEC(NVEC)=IW                                                   ALL00720
  120 CONTINUE                                                          ALL00730
      IF(I.EQ.1) NASM=NVEC                                              ALL00740
      KMASK=102                                                         ALL00750
  125 CONTINUE                                                          ALL00760
      NAAM=NVEC-NASM                                                    ALL00770
      NDET=0                                                            ALL00780
      DO 220 ITGT=1,NTU                                                 ALL00790
      KSEG=KTSEG(ITGT)                                                  ALL00800
      IF(KSEG.EQ.0) GO TO 220                                           ALL00810
      NTSTAT=ITSTAT(ITGT)                                               ALL00820
      IF(NTSTAT.LE.0) GO TO 220                                         ALL00830
      KT=MOD(KTUK(2,ITGT)/1000000,10)                                   ALL00840
      IF(NTSTAT-2 .GE. MXGAGE(KT)) GO TO 220                            ALL00850
      IVAL=0                                                            ALL00860
      IF(KHVT(ITGT)/1000 .GT. 0) IVAL=100                               ALL00870
      ISEG=KSEG/IPK                                                     ALL00880
  205 IF(ISEG.EQ.0) GO TO 220                                           ALL00890
      IPD=INFORM(ISEG+5)                                                ALL00900
      IF(IPD.GT.0) GO TO 214                                            ALL00910
  210 ISEG=MOD(INFORM(ISEG+2),IPK)                                      ALL00920
      GO TO 205                                                         ALL00930
  214 IF(NDET.LT.220) GO TO 215                                         ALL00940
      WRITE(N6,6214)                                                    ALL00950
 6214 FORMAT('  *** SUBROUTINE ALLOCA, DETECTION ARRAY LDET FULL ***')  ALL00960
      GO TO 224                                                         ALL00970
  215 NDET=NDET+1                                                       ALL00980
      IPD=MOD(IPD,100)+IVAL                                             ALL00990
      LDET(NDET)=IPD*10000+ISEG
      GO TO 210
  220 CONTINUE
      IF(NDET.EQ.0) GO TO 400
  224 ASSIGN 226 TO ISWVEC
      IVEC=0
  225 GO TO ISWVEC,(226,232,237)
  226 MAX=LDET(1)/10000
      IMAX=1
      DO 230 I=1,NDET
      K=LDET(I)/10000
      IF(K.LE.MAX) GO TO 230
      MAX=K
      IMAX=I
  230 CONTINUE
      IF(MAX.EQ.0) GO TO 400
      IPD=MAX
      IVAL=MAX/100
      MAX=LDET(IMAX)
      LDET(IMAX)=0
      ISEG=MOD(MAX,10000)
      KSEG=INFORM(ISEG)
      ITGT=MOD(KSEG,IPK)
      KT=MOD(KTUK(2,ITGT)/1000000,10)
      NTSTAT=ITSTAT(ITGT)
      MGAGE=MXGAGE(KT)+2
      IF(NTSTAT.GE.MGAGE) GO TO 225
      TX=TXYZ(ITGT,2)
      TY=TXYZ(ITGT,3)
      TZ=TXYZ(ITGT,4)
      IWPN=KSEG/IPK
      NWSTAT=IWSTAT(IWPN)
      IF(NWSTAT.LE.0) GO TO 225
  231 IS=LSYSW(IWPN)
      IPLAT=MOD(IS,100)
      IS=IS/100
      ISB=MOD(IS,10000)
      ISA=IS/10000
      MASK=LPRSYS(ISB)
      ISS=ISB-3
      KW=MOD(MASK,10)
      IF(IVEC.EQ.1) GO TO 239
      IVEC=MOD(MASK/10000000,10)
      IF(IVEC.EQ.0) GO TO 239
      ISEG=0
      IF(KT.EQ.NAIR) GO TO 234
      KCM=10
      ICM=ISA
      ICMB=ISS
      ASSIGN 232 TO ISWVEC
  232 ICM=ICM+1
      IF(NTSTAT.GE.MGAGE) GO TO 224
      IF(ICM.GT.ICMB) GO TO 233
      ISYS=LBRSYS(ICM)
      IF(MOD(ISYS,100) .NE. KCM) GO TO 232
      IWPN=MOD(ISYS/100,100)
      NWSTAT=IWSTAT(IWPN)
      IF(NWSTAT.LE.0) GO TO 232
      CALL NQUIRE(INFORM,KWSEG,IWPN,ITGT,ISEG)
      GO TO 231
  233 IF(NASM.EQ.0) GO TO 224
      IAC=1
      IBC=NASM
      GO TO 235
  234 IF(NAAM.EQ.0) GO TO 224
      IAC=NASM+1
      IBC=NVEC
  235 IF(NTSTAT.GE.MGAGE) GO TO 224
      ASSIGN 237 TO ISWVEC
      DO 236 I=IAC,IBC
      IWPN=LACVEC(I)
      WX=WXYZ(IWPN,2)
      WY=WXYZ(IWPN,3)
      DX=TX-WX
      DY=TY-WY
      GR2=DX*DX+DY*DY
  236 IWORK(I)=GR2
  237 IF(NTSTAT.GE.MGAGE) GO TO 224
      MIN=IWORK(IAC)
      IMIN=IAC
      DO 238 I=IAC,IBC
      K=IWORK(I)
      IF(K.GE.MIN) GO TO 238
      IMIN=I
      MIN=K
  238 CONTINUE
      IF(MIN.EQ.MAXX) GO TO 224
      IWORK(IMIN)=MAXX
      IWPN=LACVEC(IMIN)
      NWSTAT=IWSTAT(IWPN)
      IF(NWSTAT.LE.0) GO TO 237
      GO TO 231
  239 IF(ISEG.GT.0 .AND. INFORM(ISEG+6).GT.0) GO TO 225
      WX=WXYZ(IWPN,2)
      WY=WXYZ(IWPN,3)
      WZ=WXYZ(IWPN,4)
      DX=TX-WX
      DY=TY-WY
      GR2=DX*DX+DY*DY
      GR=SQRT(GR2)
      IF(KT-NAIR) 240,270,265
  240 ASSIGN 247 TO ISWT3
      ASSIGN 285 TO ISWT1
      ASSIGN 242 TO ISWT4
      IF(NHVT.EQ.0) GO TO 241
      IF(IVAL.GT.0) GO TO 241
      IG=KWUK(3,IWPN)
      IHVT=MOD(KHVT(IG),100)
      IF(ITSTAT(IHVT).NE.2) GO TO 241
      X=MOD(LBRSYS(ISB-1)/100,10000)
      IF(.5*X.LT.GR) GO TO 225                                          ALL02100
  241 CONTINUE                                                          ALL02110
      KSYS=12                                                           ALL02120
      IAC=4                                                             ALL02130
      GO TO 280                                                         ALL02140
  242 NSALVO=DATASM( 5,ICOL) + .5                                       ALL02150
      ISALVO=0                                                          ALL02160
      ASSIGN 243 TO ISWT2                                               ALL02170
      IRP=IROW                                                          ALL02180
      IRPD=99                                                           ALL02190
      GO TO 305                                                         ALL02200
  243 ISALVO=ISALVO+1                                                   ALL02210
      DATASM(22,ICOL)=DATASM(22,ICOL)+1.                                ALL02220
      IF(NLPR.GT.0) WRITE(N6,CMLNCH)                                    ALL02230
C                                                                       ALL02240
C  CRUISE MISSILE LAUNCHED                                              ALL02250
C                                                                       ALL02260
      IF(ISALVO.EQ.NSALVO) GO TO 245                                    ALL02270
      GO TO 305                                                         ALL02280
  245 TRUNIN=-TZERO                                                     ALL02290
      RTIME=DATASM( 6,ICOL)                                             ALL02300
      GO TO 330                                                         ALL02310
  247 IF(KW.EQ.NAIR) GO TO 255                                          ALL02320
  248 KSYS=16                                                           ALL02330
      ASSIGN 225 TO ISWT3                                               ALL02340
  249 ASSIGN 250 TO ISWT1                                               ALL02350
      GO TO 280                                                         ALL02360
  250 RMAX=WPNDAT( 1,ICOL)                                              ALL02370
      IF(GR.GT.RMAX) GO TO 282                                          ALL02380
      RTIME=WPNDAT( 6,ICOL)                                             ALL02390
      TRUNIN=GR/WPNDAT(2,ICOL)                                          ALL02400
      IF(NLPR.GT.0) WRITE(N6,SSWPN)                                     ALL02410
      GO TO 330                                                         ALL02420
  255 KSYS=22                                                           ALL02430
      ASSIGN 285 TO ISWT1                                               ALL02440
  256 IF(NTSTAT.GE.MGAGE) GO TO 225                                     ALL02450
      ASSIGN 260 TO ISWT3                                               ALL02460
      ASSIGN 305 TO ISWT4                                               ALL02470
      ASSIGN 262 TO ISWT2                                               ALL02480
  257 IRP=IPLAT                                                         ALL02490
      IAC=2                                                             ALL02500
      Z=.5*(WZ+TZ)                                                      ALL02510
      BZ=WPX1(7,IPLAT)/6.080                                            ALL02520
      VA=WPX1(NVSEA,IPLAT)                                              ALL02530
      VB=WPX1(NVALT,IPLAT)                                              ALL02540
      WV=VA+(VB-VA)*Z/BZ                                                ALL02550
      IF(WV.GT.VB) WV=VB                                                ALL02560
      LCAR=LBRSYS(ISB-2)                                                ALL02570
      ICAR=IABS(LCAR/1000000)                                           ALL02580
      IF(LCAR.LE.0) GO TO 280                                           ALL02590
      IF(IWSTAT(ICAR).GT.0) GO TO 259                                   ALL02600
  258 LCAR=-LCAR                                                        ALL02610
      LBRSYS(ISB-2)=LCAR                                                ALL02620
      GO TO 280                                                         ALL02630
  259 IF(LBRSYS(MOD(LCAR/100,10000)).EQ.0) GO TO 258                    ALL02640
```

```
      CX=WXYZ(ICAR,2)
      CY=WXYZ(ICAR,3)
      DX=CX-TX
      DY=CY-TY
      GCT2=DX*DX+DY*DY
      IF(GR2 .GT. 1.01*GCT2) GO TO 225
      GO TO 280
  260 KSYS=16
      ASSIGN 272 TO ISWT1
      ASSIGN 225 TO ISWT3
      GO TO 280
C
C  A/C HAS BEEN VECTORED
C
  262 JTIME=MAXX
      IF(LCAR.LE.0) GO TO 264
C
      LINE=MOD(LCAR/100,10000)
      DX=CX-WX
      DY=CY-WY
      RESTOR=SQRT(DX*DX+DY*DY)/WV
      IF(RESTOR.LT.RSTMIN) RESTOR=RSTMIN
      JTIME=RESTOR*1000.
      LS=LBRSYS(LINE)
      IF(LS/1000000.EQ.0) GO TO 263
      KSB=MOD(LSYSW(ICAR)/100,10000)-2
      IF(LBRSYS(KSB).GT.ITIME*100) GO TO 263
      LBRSYS(KSB)=(ITIME+LRATE)*100
      LBRSYS(LINE)=LS-1000000
      JTIME=JTIME+ITIME
      GO TO 264
  263 JTIME=JTIME+LINE*100000
  264 IWSTAT(IWPN)=-JTIME
      IF(NLPR.GT.0) WRITE(N6,ACVECT)
      GO TO 335
  265 IF(NTSTAT.GE.MGAGE) GO TO 225
      ASSIGN 272 TO ISWT1
      KSYS=24
      IF(KW.EQ.NAIR) GO TO 256
      ASSIGN 248 TO ISWT3
      GO TO 249
  270 IF(NTSTAT.GE.MGAGE) GO TO 225
      IF(KW.NE.NAIR) GO TO 225
      IF(MOD(MASK/10,1000).NE. 010) GO TO 225
      KSYS=23
      ASSIGN 225 TO ISWT3
      ASSIGN 272 TO ISWT1
      GO TO 257
  272 IV=0
      NS=WPNDAT(1,ICOL)*10.+.5
      IF(NS.GT.999) NS=999
      ITG=KTUK(3,ITGT)
      LEG=IABS(INTVLT(ITG))
      TVX=TGCV(LEG,1,ITG)
      TVY=TGCV(LEG,2,ITG)
      CALL  INCEPT(TX,TY,TVX,TVY,WX,WY,WV,WVX,WVY,DELTI)
      IF(DELTI.LE.0.) GO TO 225
      IF(DELTI.GT.AIRCPT) GO TO 225
      ASSIGN 276 TO ISWT2
      GO TO 305
  276 TUWXYZ(4,ITU)=WVX
      TUWXYZ(5,ITU)=WVY
      GO TO 262
  280 IS=ISA-1
  282 IS=IS+1
      IF(IS.GT.ISS) GO TO ISWT3,(247,225,248,260)
      ISYS=LBRSYS(IS)
      IF(MOD(ISYS,100) .NE. KSYS) GO TO 282
      IAVAIL=ISYS/10000000
      ISTOCK=MOD(ISYS/1000000,100)
      IF(ISTOCK*IAVAIL .EQ. 0) GO TO 282
      ICOL=MOD(ISYS/10000,100)
      IROW=MOD(ISYS/100,100)
      GO TO ISWT1, (285,250,272)
  285 IV=1
      IOFF=0
  290 RMAX=TTRAJ(IOFF+3,ICOL)
      IF(GR.GT.RMAX) GO TO 282
      RMIN=TTRAJ(IOFF+4,ICOL)
      IF(GR.GT.RMIN) GO TO 300
      IF(IV.EQ.2) GO TO 282
      IV=2
      IOFF=25
      GO TO 290
  300 NS=RMAX-GR
      GO TO ISWT4,(242,305)
  305 IF(NEXTU.EQ.0) GO TO 310
      ITU=NEXTU
      NEXTU=IWTU(ITU)
      GO TO 315
  310 GO TO KSWT1, (311,282)
  311 IF(LASTU.LT.MAXTU) GO TO 312
      WRITE(N6,6310)
 6310 FORMAT(   .6X,'TEMPORARY UNIT MATRIX FULL')
      ASSIGN 282 TO KSWT1
      IF(IAC.EQ.4 .AND. ISALVO.GT.0) GO TO 245
      GO TO 282
  312 IF(IAC.EQ.2) GO TO 314
      IF(ISEG.GT.0) GO TO 314
      CALL ADDSEG(INFORM,KWSEG,KTSEG,IWPN,ITGT,ISEG)
      IF(ISEG.EQ.0) GO TO 225
  314 LASTU=LASTU+1
      ITU=LASTU
  315 TUWXYZ(1,ITU)=WX
      TUWXYZ(2,ITU)=WY
      TUWXYZ(3,ITU)=WZ
      TUWXYZ(7,ITU)=TIMAX
      TUWXYZ(8,ITU)=TRACK
      IWPNTU=ITU+KWU
      IWSTAT(IWPNTU)=1
      IWTU(ITU)=ITGT*100000+IRP*1000+ICOL*10+IAC                        ALL03750
      I=GR*10.+.5                                                       ALL03760
      IF(IAC.EQ.2) I=ICAR                                               ALL03770
      IF(IAC.EQ.2) IRPD=IROW+ISTOCK*100                                 ALL03780
      IWTU(MAXTU+ITU)=   I*100+IWPN+IRPD*1000000                        ALL03790
      INVWTU(ITU)=ITIME*100000+NS*100+IV                                ALL03800
      GO TO ISWT2,(243,262,276)                                         ALL03810
  330 JTIME=(TZERO+RTIME)*1000.                                         ALL03820
      KTIME=(TZERO+TRUNIN)*1000.                                        ALL03830
      IF(JTIME.LT.KTIME) JTIME=KTIME                                    ALL03840
      IF(ISEG.EQ.0) CALL ADDSEG(INFORM,KWSEG,KTSEG,IWPN,ITGT,ISEG)      ALL03850
      IF(ISEG.EQ.0) GO TO 225                                           ALL03860
      INFORM(ISEG+3)=GR*10.+.5                                          ALL03870
      INFORM(ISEG+6)=IS*100000+JTIME                                    ALL03880
      INFORM(ISEG+7)=ITIME*100000+KTIME                                 ALL03890
      K=1000000                                                         ALL03900
      IF(ISTOCK.EQ.99) K=0                                              ALL03910
      LBRSYS(IS)=ISYS-100000000-K                                       ALL03920
  335 NTSTAT=NTSTAT+1                                                   ALL03930
      ITSTAT(ITGT)=NTSTAT                                               ALL03940
      GO TO 225                                                         ALL03950
  400 CONTINUE                                                          ALL03960
      ASSIGN 477 TO KCHEK                                               ALL03970
      IF(MOD(ICHEK,10).GT.0) ASSIGN 4765 TO KCHEK                       ALL03980
      ASSIGN 495 TO LJSET                                               ALL03990
      ASSIGN 520 TO KJSET                                               ALL04000
      IF(NTTU.EQ.0) GO TO 600                                           ALL04010
      NDET=0                                                            ALL04020
      DO 420 I=1,NTTU                                                   ALL04030
      IF(INVTTU(I).EQ.0) GO TO 420                                      ALL04040
      IF(TUTXYZ(3,I).LE.0.) GO TO 420                                   ALL04050
      K=I+NTU                                                           ALL04060
      IF(ITSTAT(K).GE.MXGAGE(4)+2 ) GO TO 420                           ALL04070
      KSEG=KTSEG(K)                                                     ALL04080
      IF(KSEG.EQ.0) GO TO 420                                           ALL04090
      ISEG=KSEG/IPK                                                     ALL04100
      T=TUTXYZ(7,I)                                                     ALL04110
      JTIME=(T-TBEGIN)*1000.                                            ALL04120
      IF(JTIME.LT.0) GO TO 420                                          ALL04130
      IF(ITTU(I)/100000.EQ.0) JTIME=22000                               ALL04140
  405 IF(ISEG.EQ.0) GO TO 420                                           ALL04150
      IPD=INFORM(ISEG+5)                                                ALL04160
      IF(IPD.LE.0) GO TO 415                                            ALL04170
      NDET=NDET+1                                                       ALL04180
      LDET(NDET)=JTIME*10000+K                                          ALL04190
      GO TO 420                                                         ALL04200
  415 ISEG=MOD(INFORM(ISEG+2),IPK)                                      ALL04210
      GO TO 405                                                         ALL04220
  420 CONTINUE                                                          ALL04230
      IF(NDET.EQ.0) GO TO 600                                           ALL04240
  425 MIN=LDET(1)                                                       ALL04250
      IMIN=1                                                            ALL04260
      DO 430 I=1,NDET                                                   ALL04270
      K=LDET(I)                                                         ALL04280
      IF(K.GE.MIN) GO TO 430                                            ALL04290
      MIN=K
      IMIN=I
  430 CONTINUE
      IF(MIN.EQ.MAXX) GO TO 600
      LDET(IMIN)=MAXX
      ITGT=MOD(MIN,10000)
      ITU=ITGT-NTU
      KSEG=KTSEG(ITGT)
      ISEG=KSEG/IPK
      MSHIP=0
      MGRP=0
      TX=TUTXYZ(1,ITU)
      TY=TUTXYZ(2,ITU)
      TZ=TUTXYZ(3,ITU)
      TVX=TUTXYZ(4,ITU)
      TVY=TUTXYZ(5,ITU)
      TVZ=TUTXYZ(6,ITU)
      TVEC=TUTXYZ(8,ITU)
      TLOS=SQRT(TZ*(6880.+TZ))
  435 IF(ISEG.EQ.0) GO TO 440
      IF(INFORM(ISEG+5).LE.0) GO TO 437
      KSEG=INFORM(ISEG)
      IWPN=KSEG/IPK
      IG=KWUK(3,IWPN)
      IF(MGRP.EQ.0) GO TO 4365
      DO 436 I=1,MGRP
      IF(LGRP(I).EQ.IG) GO TO 437
  436 CONTINUE
 4365 MGRP=MGRP+1
      LGRP(MGRP)=IG
  437 ISEG=MOD(INFORM(ISEG+2),IPK)
      GO TO 435
  440 IF(MGRP.EQ.0) GO TO 425
      DO 448 K=1,MGRP
      IG=LGRP(K)
      IUB=NWUG(IG)
      IUA=IUB/100
      IUB=IUA+MOD(IUB,100)-1
      DO 448 IWPN=IUA,IUB
      IF(IWSTAT(IWPN).LE.0) GO TO 448
      IS=LSYSW(IWPN)
      ISB=MOD(IS/100,10000)
      IF(ISB.EQ.0) GO TO 448
      IF(MOD(LBRSYS(ISB)/100,10).EQ.0) GO TO 448
      CALL NQUIRE(INFORM,KWSEG,IWPN,ITGT,ISEG)
      IF(ISEG.GT.0 .AND. INFORM(ISEG+6).GT.0) GO TO 448
      WX=WXYZ(IWPN,2)
      WY=WXYZ(IWPN,3)
      DX=TX-WX
      DY=TY-WY
      GR2=DX*DX+DY*DY
      RAIR=LBRSYS(ISB-1)/100000
      IF(GR2.GT.RAIR*RAIR) GO TO 448
      HMAST=0.
      KW=MOD(LBRSYS(ISB),10)
      IF(KW.EQ.2) GO TO 447
      LPLTW=MOD(IS,100)
      HMAST=WPX1(3,LPLTW)/FPNM
  447 HMAST=HMAST+WXYZ(IWPN,4)
      X=1.1547*(TLOS+SQRT(HMAST*(6880.+HMAST)))
      IF(GR2.GT.X*X) GO TO 448
      I=SQRT(GR2)
      MSHIP=MSHIP+1
      LSHIP(MSHIP)=I*1000000 +ISEG*100 +IWPN
  448 CONTINUE
  450 IF(MSHIP.EQ.0) GO TO 425
  451 MIN=LSHIP(1)
      IMIN=1
      DO 455 I=1,MSHIP
      K=LSHIP(I)
      IF(K.GE.MIN) GO TO 455
      MIN=K
      IMIN=I
  455 CONTINUE
      IF(MIN.EQ.MAXX) GO TO 425
      LSHIP(IMIN)=MAXX
      I=MIN/1000000
      ISEG=MOD(MIN/100, 10000)
      IWPN=MOD(MIN,100)
      IS=LSYSW(IWPN)
      LPLTW=MOD(IS,100)
      ISA=IS/1000000
      ISB=MOD(IS/100,10000)
      ISS=ISB-3
      ASSIGN 478 TO ISWT3C
      ASSIGN 475 TO ISWT3B
      KW=MOD(LBRSYS(ISB),10)
      IF(KW.EQ.NAIR) GO TO 456
      KSYS=33
      ASSIGN 457 TO ISWT1
      ASSIGN 468 TO ISWT2
      ASSIGN 472 TO ISWT3
      ASSIGN 480 TO ISWT4
      ASSIGN 555 TO ISWT5
      GO TO 460
  456 KSYS=23
      GO TO 458
  457 KSYS=43
  458 ASSIGN 451 TO ISWT1
      ASSIGN 466 TO ISWT2
      ASSIGN 474 TO ISWT3
      ASSIGN 549 TO ISWT4
      ASSIGN 560 TO ISWT5
  460 IS=ISA-1
  465 IS=IS+1
      IF(IS.GT.ISS) GO TO ISWT1,(457,451)
      ISYS=LBRSYS(IS)
      IF(MOD(ISYS,100).NE.KSYS) GO TO 465
      IAVAIL=ISYS/100000000
      GO TO ISWT2,(466,468)
```

```
 466 ISTOCK=MOD(ISYS/1000000,100)
     GO TO 470
 468 JSYS=LBRSYS(IS+1)
     ISTOCK=MOD(JSYS/1000000,100)
 470 IF(ISTOCK*IAVAIL.EQ.0) GO TO 465
     ICOL=MOD(ISYS/10000,100)
     GO TO ISWT3,(472,474)
 472 IRAD=ICOL
     ICOL=MOD(JSYS/10000,100)
 474 GO TO ISWT3B, (475,476,4783)
 475 WX=WXYZ(IWPN,2)
     WY=WXYZ(IWPN,3)
     WZ=WXYZ(IWPN,4)
     IF(KW.EQ.NAIR) GO TO 478
     ASSIGN 476 TO ISWT3B
 476 CONTINUE
     INC=IMISC(22)
     CALL SAMLT(TIME,ICOL,WX,WY,WZ,0.,0.,TX,TY,TZ,TVX,TVY,TVZ,
    1              INC,TA,TB,TFA,TFB,0)
     IF(INC.EQ.0) GO TO 465
     IF(TA.GT.TIME) GO TO 465
     GO TO KCHEK,(4765,477)
4765 IF(TFA+TIME .GT. TVEC) GO TO 465
 477 GO TO ISWT3C, (478,479)
 478 DX=TX-WX
     DY=TY-WY
     DZ=TZ-WZ
     GR2=DX*DX+DY*DY+DZ*DZ
     GR=SQRT(GR2)
     JTU=ITTU(ITU)
     KT=MOD(JTU,10)
     ASSIGN 479 TO ISWT3C
     IF(KW.NE.NAIR) GO TO 479
     ASSIGN 4783 TO ISWT3B
4783 IF(GR.GT.WPNDAT(1,ICOL)) GO TO 465
     IF(IMISC(22).EQ.0 .AND. ABS(DZ/GR).GT. .5) GO TO 465
     RTIME=WPNDAT(6,ICOL)
     WV=WPNDAT(2,ICOL)
     CALL INCEPT(TX,TY,TVX,TVY,WX,WY,WV,WVX,WVY,TFA)
     IF(TFA.LT.0.) GO TO 465
     GO TO 548
 479 GO TO ISWT4, (480,549)
 480 HMAST=WPX1(3,LPLTW)
     BETA=WRAD(1,IRAD)
     ALFA=WRAD(2,IRAD)
     FREQ=30000./WRAD(3,IRAD)
     BEAM=WRAD(22,IRAD)/57.3
     BEAM=.5*BEAM
     ETC(1)=WZ*EPNM+HMAST
     ETC(4)=WRAD(36,IRAD)
     IF(KT.EQ.NAIR) GO TO 485
     KCOL=MOD(JTU/10,100)
     RCS=DATASM(19,KCOL)
     ALTGT=.5*(DATASM(16,KCOL)+DATASM(17,KCOL))
     GO TO 490
```

```
  485 IPLTT=MCD(JTU/10C0,100)                                          ALL05950
      ALTGT=.5*(TPX1(1,IPLTT)+TPX1(2,IPLTT))                           ALL05960
      RCS=TPX1(4,IPLTT)                                                ALL05970
  49C GO TO LJSET,(495,501)                                            ALL05980
  495 ASSIGN 501 TO LJSET                                              ALL05990
      KSYSAV=KSYS                                                      ALL06000
      ITYPE=5                                                          ALL06010
      JAM=0                                                            ALL06020
      DO 500 IT=1,NTU                                                  ALL06030
      MS=LSYST(IT)/100                                                 ALL06040
      IF(MS.EQ.0) GO TO 500                                            ALL06050
      IA=MS/10000                                                      ALL06060
      IB=MOD(MS,10000)                                                 ALL06070
      MASK=LBRSYS(IB)/10                                               ALL06080
      IF(MOD(MASK/1000,10) .EQ. 0) GO TO 500                           ALL06090
      IB=IB-1                                                          ALL06100
      DO 5000 MS=IA,IB                                                 ALL06110
      KSYS=LBRSYS(MS)                                                  ALL06120
      IF(MOD(KSYS,10) .NE. ITYPE) GO TO 5000                           ALL06130
      IF(KSYS/1000000000+JAMON.EQ.0) GO TO 5000                        ALL06140
      IJAM=MOD(KSYS/10000,100)                                         ALL06150
      JAM=JAM+1                                                        ALL06160
      JWORK(JAM)=IT*100+IJAM                                           ALL06170
 5000 CONTINUE                                                         ALL06180
  500 CONTINUE                                                         ALL06190
      KSYS=KSYSAV                                                      ALL06200
      IF(JAM.GT.0) ASSIGN 503 TO KJSET                                 ALL06210
  501 GO TO KJSET,(503,520)                                            ALL06220
  503 JAMR=0                                                           ALL06230
      DO 510 JM=1,JAM                                                  ALL06240
      IJ=JWORK(JM)                                                     ALL06250
      IT=IJ/100                                                        ALL06260
      IJ=MOD(IJ,100)                                                   ALL06270
      DO 505 K=1,3                                                     ALL06280
      IF(FREQ.LT.AJAM(5,K,IJ)) GO TO 505                               ALL06290
      IF(FREQ.GT.AJAM(6,K,IJ)) GO TO 505                               ALL06300
      KJ=K                                                             ALL06310
      GO TO 506                                                        ALL06320
  505 CONTINUE                                                         ALL06330
      GO TO 510                                                        ALL06340
  506 XJ=TXYZ(IT,2)                                                    ALL06350
      YJ=TXYZ(IT,3)                                                    ALL06360
      DXJ=XJ-WX                                                        ALL06370
      DYJ=YJ-WY                                                        ALL06380
C                                                                      ALL06390
C                                                                      ALL06400
C                                                                      ALL06410
      ANGJ=ATAN2(DXJ,DYJ)                                              ALL06420
      SQJR=SQRT(DXJ*DXJ+DYJ*DYJ)                                       ALL06430
      JAMR=JAMR+1                                                      ALL06440
      IWORK(JAMR)=KJ*100+IJ                                            ALL06450
      WORKI(JAMR)=ANGJ                                                 ALL06460
      WORKJ(JAMR)=SQJR                                                 ALL06470
  510 CONTINUE                                                         ALL06480
      IF(JAMR.EQ.0) GO TO 520                                          ALL06490
      ANGT=ATAN2(DX,DY)
      DO 515 IJ=1,JAMR
      A=ABS(ANGT-WORKI(IJ))
      IF(A.GT.PI) A=TWOPI-A
      IF(A.GT.BEAM) GO TO 515
      IJAM=IWORK(IJ)
      ETC(2)=WORKJ(IJ)
      GO TO 525
  515 CONTINUE
  520 RDRNGE=BETA*RCS**.25
      ETC(2)=0.
      ETC(3)=0.
      IJAM=1
      KJAM=1
      GO TO 530
  525 KJAM=IJAM/100
      IJAM=MOD(IJAM,100)
      ETC(3)=2.
      GJ2=10.**(.1*AJAM(2,KJAM,IJAM))
      PJ2=AJAM(1,KJAM,IJAM)
      RDRNGE=ALFA*(500./PJ2*(1.26/GJ2)**.25)
  530 IF(RDRNGE.LT.GR) GO TO IRFAIL,(465,620)
      IF(IPRAD.GE.0) GO TO 532
      PDT=EXP(-.693*GR2/(RDRNGE*RDRNGE))
      GO TO 545
  532 VX=TVX
      VY=TVY
      DGRA=VX*DX+VY*DY
      VC=0.
      IF(GR.GT.0.) VC=DGRA/GR
      TAR(1)=ALTGT
      TAR(2)=RCS
      TAR(3)=TZ*FPNM
      TAR(4)=VC
      RANGE(1)=GR
      CALL RADAR(WRAD(1,IRAD),ENV,ETC,TAR,AJAM(1,KJAM,IJAM),TRAD,RANGE,
     1      PDTECT,ANG,DYN,ICK,ISCAN,IPRAD)
      PDT = PDTECT(ISCAN+1)
  545 IF(PDT.LT.PDTMIN) GO TO IRFAIL,(465,620)
      GO TO IRPASS, (614,546)
  546 IF(NLPR.GT.0) WRITE(N6,SAM)
C
C  SAM LAUNCHED
C
      GO TO 550
  548 IF(NLPR.GT.0) WRITE(N6,AAMDEF)
C
C  DEFENSIVE AAM LAUNCHED
C
      GO TO 551
  549 CONTINUE
      IF(NLPR.GT.0) WRITE(N6,SAGUN)
C
C  SA GUN ASSIGNED
C
```

```
  550 RTIME=SAMDAT(6,ICOL)                                                 ALL07050
  551 IF(RTIME.LT.TFA) RTIME=TFA                                           ALL07060
      IF(ISEG.EQ.0) CALL ADDSEG(INFORM,KWSEG,KTSEG,IWPN,ITGT,ISEG)         ALL07070
      IF(ISEG.EQ.0) GO TO 451                                              ALL07080
      JTIME=(TZERO+RTIME)*1000.                                            ALL07090
      INFORM(ISEG+6)=IS*100000+JTIME                                       ALL07100
      KTIME=(TZERO+TFA)*1000.                                              ALL07110
      INFORM(ISEG+7)=ITIME*100000+KTIME                                    ALL07120
      INFORM(ISEG+3)=GR*10.+.5                                             ALL07130
      K=1000000                                                            ALL07140
      IF(ISTOCK.EQ.99) K=0                                                 ALL07150
      GO TO ISWT5,(555,560)                                                ALL07160
  555 LBRSYS(IS)=ISYS-100000000                                            ALL07170
      LBRSYS(IS+1)=JSYS-K                                                  ALL07180
      GO TO 570                                                            ALL07190
  560 LBRSYS(IS)=ISYS-100000000-K                                          ALL07200
  570 ITSTAT(ITGT)=ITSTAT(ITGT)+1                                          ALL07210
      GO TO 425                                                            ALL07220
  600 CONTINUE                                                             ALL07230
C                                                                          ALL07240
C          SAM TRACK RADAR RECHECK AT TSTEP BEFORE INTERCEPT               ALL07250
C                                                                          ALL07260
      IF(ICHEK.LT.10) GO TO 700                                            ALL07270
      IF(NTTU.EQ.0) GO TO 700                                              ALL07280
      NEXT=(TZERO+TSTEP)*1000.                                             ALL07290
      ASSIGN 620 TO IRFAIL                                                 ALL07300
      ASSIGN 614 TO IRPASS                                                 ALL07310
      ITU=0                                                                ALL07320
  610 ITU=ITU+1                                                            ALL07330
      IF(ITU.GT.NTTU) GO TO 700                                            ALL07340
      IF(INVTTU(ITU).EQ.0) GO TO 610                                       ALL07350
      ITGT=ITU+NTU                                                         ALL07360
      IF(ITSTAT(ITGT).LE.2) GO TO 610                                      ALL07370
      ISEG=KTSEG(ITGT)/IPK                                                 ALL07380
  612 IF(ISEG.EQ.0) GO TO 610                                              ALL07390
      IWORD=INFORM(ISEG+7)                                                 ALL07400
      IF(IWORD.EQ.0) GO TO 614                                             ALL07410
      ITDI=MOD(IWORD,100000)                                               ALL07420
      IF(ITDI.GT.NEXT) GO TO 614                                           ALL07430
      IF(ITDI.LT.ITIME) GO TO 614                                          ALL07440
      LINE=INFORM(ISEG+6)/100000                                           ALL07450
      ISYS=LBRSYS(LINE)                                                    ALL07460
      KSYS=MOD(ISYS,100)                                                   ALL07470
      IF(KSYS.EQ.33) GO TO 616                                             ALL07480
  614 ISEG=MOD(INFORM(ISEG+2),IPK)                                         ALL07490
      GO TO 612                                                            ALL07500
  616 TX=TUTXYZ(1,ITU)                                                     ALL07510
      TY=TUTXYZ(2,ITU)                                                     ALL07520
      TZ=TUTXYZ(3,ITU)                                                     ALL07530
      TVX=TUTXYZ(4,ITU)                                                    ALL07540
      TVY=TUTXYZ(5,ITU)                                                    ALL07550
      TLOS=SQRT(TZ*(6880.+TZ))                                             ALL07560
      JTU=ITTU(ITU)                                                        ALL07570
      KT=MOD(JTU,10)                                                       ALL07580
      IWPN=INFORM(ISEG)/IPK                                                ALL07590
      NWSTAT=IWSTAT(IWPN)                                               ALL07600
      IF(NWSTAT.LE.0) GO TO 620                                         ALL07610
      WX=WXYZ(IWPN,2)                                                   ALL07620
      WY=WXYZ(IWPN,3)                                                   ALL07630
      WZ=WXYZ(IWPN,4)                                                   ALL07640
      IS=LSYSW(IWPN)                                                    ALL07650
      LPLTW=MOD(IS,100)                                                 ALL07660
      HMAST=WPX1(3,LPLTW)/FPNM+WZ                                       ALL07670
      DX=TX-WX                                                          ALL07680
      DY=TY-WY                                                          ALL07690
      DZ=TZ-WZ                                                          ALL07700
      GR2=DX*DX+DY*DY+DZ*DZ                                             ALL07710
      GR=SQRT(GR2)                                                      ALL07720
      X=1.1547*(TLOS+SQRT(HMAST*(6880.+HMAST)))                         ALL07730
      IF(GR.GT.X) GO TO 620                                             ALL07740
      IRAD=MOD(ISYS/10000,100)                                          ALL07750
      GO TO 480                                                         ALL07760
C                                                                       ALL07770
C        SAM LOST AT RECHECK                                            ALL07780
C                                                                       ALL07790
  620 INFORM(ISEC+7)=0                                                  ALL07800
      IF(NLPR.GT.0) WRITE(N6,SAMLST)                                    ALL07810
      GO TO 614                                                         ALL07820
  700 CONTINUE                                                          ALL07830
      RETURN                                                            ALL07840
      END                                                               ALL07850


       SUBROUTINE  SAMLT(T1,KL,XL,YL,ZL,HA,HB,XT,YT,ZT,U,V,W,INX,       SAML0010
     1                    TA,TB,TFA,TFB,IPRINT)                         SAML0020
C   PGM=NXX(NEM). L.D.G.        VER.3   10-5-73      FORTRAN IV    EBCD SAML0030
C   MODS.101,149,150,172,194,195,216.                  VER.3   1-25-75 SAML0040
C   TO FIND EARLIEST AND LATEST SAM LAUNCH TIMES PERMITTED.             SAML0050
C      T1      = CURRENT TIME, HRS                                      SAML0060
C      KL      = LAUNCHER TYPE INDEX, 1.LE.KL.LE.N                      SAML0070
C      XL,YL,ZL = EAST,NORTH,& VERTICAL COORD. OF  LAUNCHER, NM         SAML0080
C      XT,YT,ZT = COORD. OF TARGET, NM                                  SAML0090
C      U , V, W = VELOCITY COMPONENTS OF TARGET, KNOTS                  SAML0100
C      INX   = 0, NO FIRE CONTROL SOLUTION, TA = TB = T1                SAML0110
C      TA      = EARLIEST PERMISSIBLE LAUNCH TIME.GE.T1                 SAML0120
C      TB      = LATEST  PERMISSIBLE LAUNCH TIME.GE.TA                  SAML0130
C      IPRINT= 0, NO PRINT.    =1, PRINT.                               SAML0140
C      HA,HB = MIN AND MAX AZIMUTH LIMITS  (0,PI2)                      SAML0150
C***                                                                    SAML0160
CSAMLT               8,12,15,                                           SAML0170
       COMMON/CSAMLT/L,M,N,RMN(15),RMX(15),EMN(15),EMX(15),ZMX(15),     SAML0180
     1 XMX(15),  RT(8,12,15)                                            SAML0190
C***                                                                    SAML0200
C      L,M,N = DIMENSIONS OF  RT. THESE ARE SET IN BLOCK DATA.          SAML0210
C      KLTYP(KL)= THE CATALOG CODE OF LAUNCHER TYPE  KL.                SAML0220
C      RMN(KL),RMX(KL) = MIN & MAX RANGES,NM, OF SAM TYPE KL.           SAML0230
C      EMN(KL),EMX(KL) = MIN & MAX ELEV ANGLES                          SAML0240
C      ZMX(KL),XMX(KL) = MAX VERTICAL & HORIZONTAL SECTOR LIMITS        SAML0250
C      RT(I,J,KL) = TIME OF FLIGHT CONTOURS FOR LAUNCHER KL.            SAML0260
```

```
C***
C**STATEMENT FUNCTIONS
      SSQF(X,Y,Z)  =   X**2 + Y**2 + Z**2
      EFF (X,Y,Z)  =   ATAN2(Z,SQRT(X**2 + Y**2) )
      HFF (X,Y )   =   ATAN2(X, Y)
      DATA  PI2,RAD/6.28319,57.29578/
      NAMELIST/NAMA/KLL,TIME,X1,Y1,Z1,UU,VV,WW,R1,H1,E1,TAUZ,DZ
      NAMELIST/NAMB/TAUZ,DZ,XZ,YZ,ZZ,RZ,HZ,EZ,DESCB
      NAMELIST/NAMC/DESCB,TAURB1,TAURB2
      NAMELIST/NAMD/TAU2,X2,Y2,Z2,R2,H2,E2,INR2,INE2,TF2
      NAMELIST/NAME/TAU3,X3,Y3,Z3,R3,H3,E3,INR3,INE3,TF3
      NAMELIST/NAMF/TIME,TAA,TBB,INXX
C***
      KOPT=INX
      N6   = 6
 1    INX = 0
      KPRINT = 1
      TIME = T1
      UU = U
      VV = V
      WW = W
      KLL  = KL
      RASQ = RMN(KL)**2
      RBSQ = RMX(KL)**2
      EA    = EMN(KL) / RAD
      EB    = EMX(KL) / RAD
      ZA    = .0
      ZB    = ZMX(KL) / 6.080
      XBSQ= XMX(KL)**2
      X1    = XT - XL
      Y1    = YT - YL
      Z1    = ZT - ZL
      R1SQ = SSQF(X1,Y1,Z1)
      R1    = SQRT(R1SQ)
      E1    = EFF (X1,Y1,Z1)
      H1    = HFF (X1,Y1)
      IF (H1.LT..0)   H1=H1 + PI2
      UVSQ = U** 2 + V**2
      TAUZ = -(U*X1 + V*Y1) / UVSQ
      DZ    = -1.
C**   IS TARGET RECEDING
      IF (TAUZ.GE..0)  GO TO 10
C**   NO INTERCEPT. PRINT OPTION
 8    INX = 0
      TA    = T1
      TB    = T1
      IF (IPRINT.LE.0) RETURN
      GO TO (65,64,63) , KPRINT
C**   IS TARGET TRACK COMPLETELY OUTSIDE SECTOR
C**   LOOK AT CROSSING POINT
 10   DZ    = ABS(U*Y1 - V*X1) / SQRT(UVSQ)
      DZSQ = DZ * DZ
      IF (DZSQ.GE.RBSQ)  GO TO 8
      DESCB =   7777.
      XZ    = X1 + U*TAUZ
      IF (ABS(XZ).LE..0001)    XZ=.0001                                 SAML0820
      YZ   = Y1 + V*TAUZ                                                SAML0830
      ZZ   = Z1 + W*TAUZ                                                SAML0840
      RZSQ = SSOF(XZ,YZ,ZZ)                                             SAML0850
      EZ   = EFF (XZ,YZ,ZZ)                                             SAML0860
      HZ   = HFF (XZ,YZ)                                                SAML0870
      IF (HZ.LT..0)   HZ=HZ + PI2                                       SAML0880
      RZ   = SQRT(RZSQ)                                                 SAML0890
      KPRINT = 2                                                        SAML0900
      IF (RZSQ.GE.RBSQ)  GO TO 8                                        SAML0910
C***  LOOK AT OUTER SHELL INTERSECTIONS                                  SAML0920
 12   A  =  UVSQ + W**2                                                 SAML0930
      B  =  2.*(U*X1 + V*Y1 + W*Z1)                                     SAML0940
      CB =  R1SQ - RBSQ                                                 SAML0950
      DESCB = B**2  - 4.*A*CB                                           SAML0960
      IF (DESCB.LE..0)  GO TO 8                                         SAML0970
C***  TARGET GOES THRU OUTER SHELL. FIND TIMES.                          SAML0980
      DESC = SQRT(DESCB)                                                SAML0990
      TAURB1 = (-B - DESC) / (2.*A)                                     SAML1000
      TAURB2 = (-B + DESC) / (2.*A)                                     SAML1010
C***  IS TARGET IN OUTER SHELL LONG ENUF                                 SAML1020
      IF((TAURB2 - TAURB1).GE..0003)  GO TO 13                          SAML1030
 121  KPRINT = 3                                                        SAML1040
      GO TO 8                                                           SAML1050
C**   PREPARE FOR AZIMUTH TESTS                                         SAML1060
 13   HAA = HA                                                          SAML1070
      HBB = HB                                                          SAML1080
      IF (HAA.LT..0)  HAA = HAA + PI2                                   SAML1090
      IF (HBB.LT..0)  HBB = HBB + PI2                                   SAML1100
      HOFF = .0                                                         SAML1110
      IF (HAA.LE.HBB)  GO TO 131                                        SAML1120
      HOFF = PI2 - HAA                                                  SAML1130
      HAA  = .0                                                         SAML1140
      HBB = HBB + HOFF                                                  SAML1150
 131  ITESTH = 1                                                        SAML1160
      DHH = HBB - HAA                                                   SAML1170
      IF ((DHH.GT.(PI2-.01)).OR.(ABS(DHH).LT..01))  ITESTH=0            SAML1180
C***  IS  PZ  IN SECTOR                                                  SAML1190
 14   INZ = 0                                                           SAML1200
      IF (ITESTH.EQ.0)  GO TO 141                                       SAML1210
      HZM = AMOD(HZ + HOFF, PI2)                                        SAML1220
      IF (HZM.LT.HAA.OR.HZM.GT.HBB)  GO TO 15                           SAML1230
 141  IF ((EZ.GE.EA.AND.EZ.LE.EB).AND.(ZZ.LE.ZB).AND.                   SAML1240
     1    ((XZ**2+YZ**2).LE.XBSQ).AND.                                  SAML1250
     2    (RZSQ.GE.RASQ.AND.RZSQ.LE.RBSQ) )  INZ=1                      SAML1260
C***  IS P1 INSIDE OUTER SHELL. CHOOSE  PS21.                            SAML1270
 15   TAUS21 = .0                                                       SAML1280
      IF (TAURB1.LE..0)  GO TO 16                                       SAML1290
      TAUS21= TAURB1                                                    SAML1300
 16   XS21  = X1 + U*TAUS21                                             SAML1310
      YS21  = Y1 + V*TAUS21                                             SAML1320
      ZS21  = Z1 + W*TAUS21                                             SAML1330
      IF (ZS21.LT..0)  GO TO 121                                        SAML1340
      RS21  = SSOF(XS21,YS21,ZS21)                                      SAML1350
      ES21  = EFF (XS21,YS21,ZS21)                                      SAML1360
      HS21  = HFF (XS21,YS21 )
      IF (HS21.LT..0)   HS21=HS21 + PI2
C***  IS PS21 IN SECTOR
      INS21 = 0
      IF (ITESTH.EQ.0)  GO TO 17
      HS21M = AMOD(HS21 + HOFF, PI2)
      IF (HS21M.LT.HAA.OR.HS21M.GT.HBB)  GO TO 18
 17   IF ((ES21.GE.EA.AND.ES21.LE.EB).AND.(ZS21.LE.ZB).AND.
     1      ((XS21**2+YS21**2).LE.XBSQ).AND.
     2      (RS21.GE.RASQ.AND.RS21.LE.RBSQ) )  INS21=1
C***  SEARCH FOR TIME ENTER SECTOR
 18   IF (INS21.EQ.1)  GO TO 28
      TAUS22= TAUZ
      DTAU  = TAUS22 - TAUS21
      IF (DTAU.LT..0002)  GO TO 121
      DT    = AMAX1(DTAU/10.,.00005)
      TAUS22= TAUS22 - DT
      TAU   = TAUS21
      KS1   = 1
 20   TAU = TAU + DT
      X   = X1 + U*TAU
      Y   = Y1 + V*TAU
      Z   = Z1 + W*TAU
      IF (Z.LT..0.OR.Z.GT.ZB)  GO TO 22
      IF ((X**2+Y**2).GT.XBSQ)  GO TO 22
      RSQ = SSQF(X,Y,Z)
      E   = EFF (X,Y,Z)
C***  TEST POINT
      IF (ITESTH.EQ.0)  GO TO 21
      H   = HFF(X,Y)
      IF (H.LT..0)    H=H + PI2
      HM  = AMOD(H + HOFF, PI2)
      IF (HM.LT.HAA.OR.HM.GT.HBB)  GO TO 22
 21   IF ((E.GE.EA.AND.E.LE.EB).AND.(RSQ.GE.RASQ))  GO TO 24
 22   IF (TAU.LT.TAUS22)  GO TO 20
      IF (KS1.GE.2)       GO TO 26
      IF (INZ.EQ.0)       GO TO 121
      IF (DT.LT..00005)  GO TO 121
      TAUS22= TAUS22 + DT
      DT    = 0.1 * DT
      TAUS22= TAUS22 - DT
      GO TO 20
C***  BOUNDED. DROP STEP SIZE. FIND ENTER SECTOR
 24   IF (KS1.GE.2)  GO TO 26
      TAUS22= TAU
      TAU   = TAU - DT
      DT    = 0.1 * DT
      TAUS22= TAUS22 - DT
      KS1 = 2
      GO TO 20
C***  HERE IS THE TIME THE TARGET ENTERS THE SECTOR
 26   TAU2 = TAU
      GO TO 40
 28   TAU2 = TAUS21
C***  SEARCH FOR TIME LEAVE SECTOR
```

```
 40     IF ( INZ.EQ.1)  GO TO 50
        TAUS31 = TAU2
        TAUS32 = TAUZ
        DTAU = TAUS32 - TAUS31
        IF (DTAU.LE..0002)  GO TO 121
        DT    = AMAX1(DTAU/10.,.00005)
        TAUS32= TAUS32 - DT
        TAU    = TAUS31
        KS2    = 1
 42     TAU  = TAU + DT
        X    = X1 + U*TAU
        Y    = Y1 + V*TAU
        Z    = Z1 + W*TAU
        IF (Z.LT..0.OR.Z.GT.ZB)  GO TO 46
        RSQ  = SSQF(X,Y,Z)
        E    = EFF (X,Y,Z)
C**    TEST POINT
        IF (ITESTH.EQ.0)  GO TO 43
        H    = HFF(X,Y)
        IF (H.LT..0)   H=H + PI2
        HM   = AMOD(H + HOFF, PI2)
        IF (HM.LT.HAA.OR.HM.GT.HBB)  GO TO 46
 43     IF ((E.GE.EA.AND.E.LE.EB).AND.(RSQ.GE.RASQ))  GO TO 44
        GO TO 46
 44     IF (TAU.LT.TAUS32)  GO TO 42
        IF (KS2.GE.2)       GO TO 48
        IF (DT.LT..00005)  GO TO 50
        TAUS32= TAUS32 + DT
        DT    = 0.1 * DT
        TAUS32= TAUS32 - DT
        GO TO 42
C**    BOUNDED.  DROP STEP SIZE. FIND LEAVE SECTOR.
 46     IF (KS2.GE.2)  GO TO 48
        TAUS32= TAU
        TAU   = TAU - DT
        DT    = 0.1 * DT
        TAUS32= TAUS32 - DT
        KS2   = 2
        GO TO 42
C***   HERE IS THE TIME THE TARGET LEAVES THE SECTOR, OR REACHES THE
C        CROSSING POINT, OR THE LAUNCHER.
 48     TAU3  = TAU
        GO TO 54
 50     TAU3  = TAUZ
C***   GET TIME OF FLIGHT AT POINT  P2 .
 54     X2   = X1 + U*TAU2
        Y2   = Y1 + V*TAU2
        Z2   = Z1 + W*TAU2
        R2   = SQRT(SSQF(X2,Y2,Z2) )
        E2   = EFF (X2,Y2,Z2) * RAD
        CALL  CONINT(RT,L,M,N,KL,R2,E2,TF2,INR2,INE2)
        TF2 = TF2 / 3600.
        TFA = TF2
        TA   = AMAX1(T1+TAU2-TF2, T1)
C***   GET TIME OF FLIGHT AT POINT  P3 .
```

```
      X3  = X1 + U*TAU3                                                 SAML2470
      Y3  = Y1 + V*TAU3                                                 SAML2480
      Z3  = Z1 + W*TAU3                                                 SAML2490
      R3  = SQRT(SSQF(X3,Y3,Z3) )                                       SAML2500
      E3  = EFF (X3,Y3,Z3) * RAD                                        SAML2510
      CALL  CONINT(RT,L,M,N,KL,R3,E3,TF3,INR3,INE3)                     SAML2520
      TF3 = TF3 / 3600.                                                 SAML2530
      TFB = TF3                                                         SAML2540
      TB  = AMAX1(T1+TAU3-TF3, T1, TA)                                  SAML2550
      IF (TA.LT.TB)  INX=1                                              SAML2560
C                                                                       SAML2570
C**  INTERPOLATE INTERCEPT FLIGHT TIME(TFA) WHEN TA = CURRENT TIME(T1)  SAML2580
C                                                                       SAML2590
      IF(INX.EQ.0) GO TO 59                                             SAML2600
      IF(TAU2.GE.TFA) GO TO 59                                          SAML2610
      IF(KOPT.NE.0) GO TO 59                                            SAML2620
      LOOP=0                                                            SAML2630
 56   TCEPT=(TAU2+TAU3)*.5                                              SAML2640
      X2=X1+U*TCEPT                                                     SAML2650
      Y2=Y1+V*TCEPT                                                     SAML2660
      Z2=Z1+W*TCEPT                                                     SAML2670
      R2=SQRT(SSQF(X2,Y2,Z2))                                           SAML2680
      E2=EFF(X2,Y2,Z2)*RAD                                              SAML2690
      CALL CONINT(RT,L,M,N,KL,R2,E2,TFA,INR2,INE2)                      SAML2700
      TFA=TFA/3600.                                                     SAML2710
      DELT=TFA-TCEPT                                                    SAML2720
      IF(ABS(DELT).LE. .0002) GO TO 59                                  SAML2730
      LOOP=LOOP+1                                                       SAML2740
      IF(LOOP.GE.10) GO TO 59                                           SAML2750
      IF(DELT) 57,59,58                                                 SAML2760
 57   TAU3=TCEPT                                                        SAML2770
      GO TO 56                                                          SAML2780
 58   TAU2=TCEPT                                                        SAML2790
      GO TO 56                                                          SAML2800
 59   CONTINUE                                                          SAML2810
      IF (IPRINT.LE.0)  GO TO 70                                        SAML2820
C**   PRINT OPTIONS                                                     SAML2830
 60   TAA = TA                                                          SAML2840
      TBB = TB                                                          SAML2850
      INXX = INX                                                        SAML2860
      WRITE(N6,NAMF)                                                    SAML2870
 61   IF (IPRINT.LE.1)  GO TO 65                                        SAML2880
      H3  = HFF(X3,Y3) * RAD                                            SAML2890
      IF (H3.LT..0)   H3=H3 + 360.                                      SAML2900
      WRITE(N6,NAME)                                                    SAML2910
 62   H2  = HFF(X2,Y2) * RAD                                            SAML2920
      IF (H2.LT..0)   H2=H2 + 360.                                      SAML2930
      WRITE(N6,NAMD)                                                    SAML2940
 63   IF (IPRINT.LE.2)  GO TO 64                                        SAML2950
      WRITE(N6,NAMC)                                                    SAML2960
 64   IF (IPRINT.LE.1)  GO TO 65                                        SAML2970
      HZ  = HZ * RAD                                                    SAML2980
      EZ  = EZ * RAD                                                    SAML2990
      WRITE(N6,NAMB)                                                    SAML3000
 65   IF (IPRINT.LE.0)  GO TO 70                                        SAML3010
```

```
      H1  = H1 * RAD                                                    SAML3020
      E1  = E1 * RAD                                                    SAML3030
      WRITE(N6,NAMA)                                                    SAML3040
   70 RETURN                                                            SAML3050
      END                                                               SAML3060


      SUBROUTINE  CONINT(RT,L,M,N,KN,R,A,Z,INR,INA)                     CINT0010
C  PGM=NXX. L.D.G.           VER.1  8-11-73          FORTRAN IV     EBCD CINT0020
C  TO INTERPOLATE BETWEEN CONTOURS. 3-DIM. STORAGE OF 2-WAY TABLES.     CINT0030
C  BORDERED TABLES. 1ST COL= CONTOUR VALUES, Z                          CINT0040
C                       1ST ROW= ANGLE IN POLAR COORD., A.              CINT0050
C                       CORNER = 100*M + N,  MATRIX M X N INCL BORDERS. CINT0060
C  TABLED VALUES = POLAR LENGTH TO CONTOUR, RT.                         CINT0070
C  ASSUMES, (1) CONTOURS ARE SINGLE VALUED WITH RESPECT TO ANGLE 'A'.   CINT0080
C           (2) BORDER VALUES MUST INCREASE.                            CINT0090
C           (3) IF OUTSIDE INTERVAL, USES NEAREST VALUE                 CINT0100
C           (4) IF R.LT.MIN, INR=-1.  IF R.GT.MAX, INR=+1.              CINT0110
C           (5) IF A.LT.MIN, INA=-1.  IF A.GT.MAX, INA=+1.              CINT0120
C  RETURNS  Z = INTERPOLATED CONTOUR VALUE FOR POINT (A,R)              CINT0130
C***                                                                    CINT0140
C  NOTE. SUBROUTINE WORKS EQUALLY WELL FOR CONTOURS DEFINED IN          CINT0150
C       RECTANGULAR  X VS Y  COORD. E.G. LET X=A, Y=R.                  CINT0160
C***                                                                    CINT0170
      DIMENSION  RT(L,M,N)                                              CINT0180
    1 INR  = 0                                                          CINT0190
      INA  = 0                                                          CINT0200
    2 IM   = RT(1,1,KN) + 0.1                                           CINT0210
      IN   = MOD(IM,100)                                                CINT0220
      IM   = IM / 100                                                   CINT0230
C***  CONSIDER ANGLE  A                                                 CINT0240
      AMX = RT(1,IN,KN)                                                 CINT0250
      AMN = RT(1, 2,KN)                                                 CINT0260
      IF (A - AMX)  6, 4, 4                                             CINT0270
    4 INA  = 1                                                          CINT0280
      P    = 0.                                                         CINT0290
      Q    = 1.                                                         CINT0300
      IA   = IN - 1                                                     CINT0310
      GO TO 16                                                          CINT0320
    6 IF (AMN - A) 10, 8, 8                                             CINT0330
    8 INA  = -1                                                         CINT0340
      P    = 1.                                                         CINT0350
      Q    = 0.                                                         CINT0360
      IA   = 2                                                          CINT0370
      GO TO 16                                                          CINT0380
C***  SEARCH FOR ANGLE BOUNDS                                           CINT0390
   10 DO 12 J=3,IN                                                      CINT0400
      IF (RT(1,J,KN) - A) 12,14,14                                      CINT0410
   12 CONTINUE                                                          CINT0420
      GO TO 4                                                           CINT0430
   14 IA   = J - 1                                                      CINT0440
      A1   = RT(1,IA,KN)                                                CINT0450
      A2   = RT(1,IA+1,KN)                                              CINT0460
      P    = (A-A2) / (A1-A2)                                           CINT0470
      Q   = 1.0 - P                                                 CINT0480
C***  CONSIDER RADIUS  R                                            CINT0490
 16   RMN = P*RT(2,IA,KN) + Q*RT(2,IA+1,KN)                         CINT0500
      IF (RMN - R) 20,18,18                                         CINT0510
 18   INR = -1                                                      CINT0520
      S  = 1.                                                       CINT0530
      T   = 0.                                                      CINT0540
      IR  = 2                                                       CINT0550
      GO TO 26                                                      CINT0560
 20   R1  = RMN                                                     CINT0570
      DO 22 I=3, IM                                                 CINT0580
      R2  = P*RT(I,IA,KN) + Q*RT(I,IA+1,KN)                         CINT0590
      IF (R2 - R) 22,24,24                                          CINT0600
 22   R1  = R2                                                      CINT0610
      INR = +1                                                      CINT0620
      S   = 0.                                                      CINT0630
      T   = 1.                                                      CINT0640
      IR  = IM-1                                                    CINT0650
      GO TO 26                                                      CINT0660
 24   IR  = I - 1                                                   CINT0670
      S   = (R-R2) / (R1-R2)                                        CINT0680
      T   = 1.0 - S                                                 CINT0690
 26   Z   = S*RT(IR,1,KN) + T*RT(IR+1,1,KN)                         CINT0700
      RETURN                                                        CINT0710
      END                                                           CINT0720


      SUBROUTINE WPNTRM(KWU,KWSEG,                                  WPNT0010
     1      KTSEG,ITSTAT,KTU,LSYST,ITTU,MAXWTU,                     WPNT0020
     2      INFORM,LBRSYS)                                          WPNT0030
      COMMON/ECONST/ VAP(15),IMISC(35)                              WPNT0040
      COMMON/STRCON/  JPK,IPK                                       WPNT0050
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP      WPNT0060
      DIMENSION KWSEG(1),KTSEG(1),INFORM(1),LBRSYS(1),ITSTAT(1)     WPNT0070
      DIMENSION LSYST(1),ITTU(1)                                    WPNT0080
      DIMENSION JWP(4),WEP(5),JTG(5),TGT(5)                         WPNT0090
      DATA JWP,JTG,WEP,TGT/9*0,      10*0./                         WPNT0100
      NAMELIST/SAMGSS/ TIME,IWPN,KSYS,ITGT,KSTATE                   WPNT0110
      NLPR=IMISC(3)                                                 WPNT0120
      ISIDE=IMISC(4)                                                WPNT0130
      IPR=IMISC(5)                                                  WPNT0140
      INIT=0                                                        WPNT0150
      N6=6                                                          WPNT0160
      ITIME=(TIME-TBEGIN)*1000.                                     WPNT0170
      DO 150 IWU=1,KWU                                              WPNT0180
      KSEG=KWSEG(IWU)                                               WPNT0190
      ISEG=KSEG/IPK                                                 WPNT0200
  100 IF(ISEG.EQ.0) GO TO 150                                       WPNT0210
      IWORD=INFORM(ISEG+7)                                          WPNT0220
      IF(IWORD.EQ.0) GO TO 110                                      WPNT0230
      ITDI=MOD(IWORD,100000)                                        WPNT0240
      IF(ITDI.EQ.0) GO TO 109                                       WPNT0250
      IF(ITDI.GT.ITIME) GO TO 140                                   WPNT0260
      ITGT=MOD(INFORM(ISEG),10000)                                  WPNT0270
      IF(ITSTAT(ITGT).LE.0) GO TO 109
      LINE=INFORM(ISEG+6)/100000
      ISYS=LBRSYS(LINE)
      ITYPE=MOD(ISYS,100)
      IF(ITYPE.EQ.0) GO TO 109
      IF(ITYPE.EQ.33) ISYS=LBRSYS(LINE+1)
      JWP(1)=MOD(ISYS/100,100)
      JWP(3)=INFORM(ISEG)/10000
      JWP(4)=0
      RANGE=INFORM(ISEG+3)
      WEP(1)=RANGE*.1
      TL=IWORD/100000
      TC=ITDI
      WEP(2)=(TC-TL)*.001
      JTGT=ITGT-KTU
      IF(JTGT.GT.0) GO TO 102
      JTG(1)=1
      JTG(2)=ITGT
      JTG(4)=MOD(LSYST(ITGT),100)
      GO TO 105
  102 ITTU1=ITTU(JTGT)
      JTG(1)=MOD(ITTU1,10)/2
      JTG(2)=MOD(ITTU(MAXWTU+JTGT),100)
      IRP=MOD(ITTU1/1000,100)
      JTG(3)=IRP
      JTG(4)=IRP
  105 CONTINUE
      CALL KILLEX(INIT,ISIDE,JWP,WEP,JTG,TGT,IPR,KSTATE)
      IF(KSTATE.GT.0) ITSTAT(ITGT)=KSTATE*1000+MOD(ITSTAT(ITGT),1000)
      IF(KSTATE.EQ.0) ITSTAT(ITGT)=-1
      IF(NLPR.EQ.0) GO TO 109
      IWPN=INFORM(ISEG)/10000
      KSYS=MOD(LBRSYS(LINE),100)
      WRITE(N6,SAMGSS)
C
C  POSSIBLE INTERCEPT UNIT KILLED
C POSSIBLE PRIMARY UNIT KILLED
C
  109 CONTINUE
C
C  FINAL DISPOSITION FOR SAMS, SAGS, GENERAL WPNS
C
      INFORM(ISEG+7)=0
  110 IWORD=INFORM(ISEG+6)
      IF(IWORD.EQ.0) GO TO 115
      ITES=MOD(IWORD,100000)
      IF(ITES.GT.ITIME) GO TO 140
      LINE=IWORD/100000
      LBRSYS(LINE)=LBRSYS(LINE)+100000000
      INFORM(ISEG+6)=0
  115 IWORD=INFORM(ISEG+4)
      IF(ITIME.LT.MOD(IWORD,100000)) GO TO 140
      CALL DELSEG(INFORM,KWSEG,KTSEG,ISEG)
  140 ISEG=MOD(INFORM(ISEG+1),IPK)
      GO TO 100
  150 CONTINUE                                                           WPNT0830
      RETURN                                                             WPNT0840
      END                                                                WPNT0850


      SUBROUTINE  KILLEX(INIT,KSIDE,JWP,WEP,JTG,TGT,IPRNT,KSTAA)         KILX0010
C*** PGM=NXX(NEM)   L.D.G.      12-7-73         FORTRAN IV    EBCD      KILX0020
CMCCACC.PRINT VAR=1081,1211,1261,1460,1481,1631,1760,1781,1931. 12-6-74 KILX0030
C        NAMELIST=4740-4800,5020-5070.                                   KILX0040
C     TO INITIALIZE FOR EACH MONTE CARLO PASS. THEN KILL AND UPDATE      KILX0050
C***  WEAPONS                                                            KILX0060
      DIMENSION  JWP(4), WEP(5)                                          KILX0070
C     INIT  = 0, DO NOT INITIALIZE IN THIS CALL                          KILX0080
C     TIME  = BATTLE TIME IN HOURS                                       KILX0090
C     JSIDE = SIDE OF THE ATTACKER (1=BLU), (2=RED)                      KILX0100
C     JWP(1)= KSA, ROW NUMBER IN KBST( ) OR KRST( )                      KILX0110
C        (2)= UNUSED                                                     KILX0120
C        (3)= KUA, UNIT INDEX IN KBUK OR KRUK OF UNIT WHICH LAUNCHED     KILX0130
C        (4)= JAM, (=0,NO JAM),(=1,JAM)                                  KILX0140
C     WEP(1)= RNG, RANGE TO TARGET AT LAUCH (GUN OPENFIRE), NM           KILX0150
C        (2)= TOFG, TIME OF FLIGHT OR DURATION OF GUNFIRE                KILX0160
C                                                                        KILX0170
C***  TARGETS                                                            KILX0180
      DIMENSION  JTG(5), TGT(5)                                          KILX0190
C     JTG(1)= JPT,(=1,PERMANENT UNIT),(=2,TEMPORARY UNIT)                KILX0200
C        (2)= KU ,TARGET UNIT INDEX IN KBUK(I,KU) OR KRUK(I,KU)          KILX0210
C                 IF A PERMANENT UNIT                                    KILX0220
C        (3)= KSB, ROW NUMBER IN KBST(KSB) OR KRST(KSB) IF TEMPORARY     KILX0230
C     (4)= KPB, TARGET TYPE INDEX IN KBPT(KPB) OR KRPT(KPB)              KILX0240
C        (5)= UNUSED                                                     KILX0250
C     TGT( )= UNUSED                                                     KILX0260
CKILLF                                                                   KILX0270
      COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,          KILX0280
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULCST(100),        KILX0290
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                         KILX0300
C***                                                                     KILX0310
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12              KILX0320
      COMMON/INOUU/ IPR(16),JPAR(16), PAR(16),LABEL                      KILX0330
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                       KILX0340
     1,IDUMB(72), NFLAG, NFLAG2                                          KILX0350
CNAVIG                                                                   KILX0360
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                   KILX0370
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),       KILX0380
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),       KILX0390
     3          TTIME,NUMX,                                              KILX0400
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),    KILX0410
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)     KILX0420
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                      KILX0430
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),            KILX0440
     1   NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),         KILX0450
     2   BPX1(12,15),                                                    KILX0460
     3                NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),         KILX0470
     4   NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),         KILX0480
     5   RPX1(12,15)                                                     KILX0490
```

```
CMSLSYS                             16      2       2      6       6        24      50  KILX0500
         COMMON /CMSLSY/ NMSL,NMSLMX,NBSSM,NBASM,NRSSM,NRASM,LATMSL,LTPAJ,  KILX0510
     1      NAMMSL(2,16),KMSLTY(16),                                        KILX0520
     2      DATMSL(24,16), TTRAJ(50,16)                                     KILX0530
CZZSYS                              25        11        12    15            KILX0540
         COMMON/CZZSYS/NZZSYS,NZZSMX,NBZSYS,NRZSYS,LAZZ,NAMSYS(2,25),       KILX0550
     1      KZZTYP(25),ZZSYS(15,25)                                         KILX0560
CSAMLT               8,12,15,                                               KILX0570
         COMMON/CSAMLT/L,M,N,RMN(15),RXM(15),EMN(15),EMX(15),ZMX(15),       KILX0580
     1 XMX(15),    RT(8,12,15)                                              KILX0590
CSASYS                              15        4        4        2       4       20  KILX0600
         COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,        KILX0610
     1      NAMSA(2,15),KSATYP(15),SASYS(20,15)                             KILX0620
CRADAR                                                                      KILX0630
         COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                           KILX0640
     1     BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,3,2),           KILX0650
     2                   NRSR,NRTR,NRRMX,NRJ,NRJMX,                         KILX0660
     3     RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,3,2)            KILX0670
CHOMER                                                                      KILX0680
         COMMON/CHOMER/ NHOM,NHOMMX,NAMHOM(2,10),KHOMTY(10),DATHOM(36,10)   KILX0690
         COMMON/ETIME/TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP            KILX0700
 2001    FORMAT(1HO,'ERROR IN KILLEX. E1=',F6.2,',   TIME=',F10.4,          KILX0710
     1          ',   SIDE=',I3/                                             KILX0720
     2     6X,'KWEPTY=',I10,',     KSA,KUA,JAM,RNG=',3I5,F8.2/              KILX0730
     3     6X,'KTGTTY=',I10,',     JPT,KU ,KSB,KPB=',4I5)                   KILX0740
C***                                                                        KILX0750
 1       JSIDE = KSIDE                                                      KILX0760
         IPRINT = IPRNT                                                     KILX0770
         KSTAA = -1                                                         KILX0780
         IF (INIT.EQ.0)     GO TO 20                                        KILX0790
C***    INITIALIZE                                                          KILX0800
         INIT = 0                                                           KILX0810
         NUNMX = NUMX + NUMX                                                KILX0820
         NUN   = KBU + KRU                                                  KILX0830
         NSYTMX = NBSSMX + NRSSMX                                           KILX0840
         NSYT = NBSS + NRSS                                                 KILX0850
C        IF (VULFS*VULFA*VULFM.GT..00001)     GO TO 8                       KILX0860
         VULFS = .5                                                         KILX0870
         VULFA = .1                                                         KILX0880
         VULFM = .1                                                         KILX0890
 8       DO 10 I=1,NUNMX                                                    KILX0900
         KSTATE(I) = 5                                                      KILX0910
         PPROD(I) = 1.                                                      KILX0920
         PKLAST(I) = .0                                                     KILX0930
         CUMWT(I) = .0                                                      KILX0940
         VULOST(I) = .0                                                     KILX0950
 10      VUKILL(I) = .0                                                     KILX0960
         DO 12 I=1,NSYTMX                                                   KILX0970
 12      VWKILL(I) = .0                                                     KILX0980
         RETURN                                                             KILX0990
C***    SETUP TO EVALUATE KILL                                              KILX1000
 20      KSA = JWP(1)                                                       KILX1010
         KPA = JWP(2)                                                       KILX1020
         KUA = JWP(3)                                                       KILX1030
         JAM = JWP(4)                                                       KILX1040
```

```
      RNG = WEP(1)
      TOFG= WEP(2)
      JPT = JTG(1)
      KU  = JTG(2)
      KSB = JTG(3)
      KPB = JTG(4)
      KWEAP = KSA
      KWUNI = KUA
      KTGTU = KU
      KTGTTY = 0
      KWEPTY = 0
      E1 = 20.
      GO TO (24,26), JSIDE
C***  ERROR MESSAGE
 22    IF (IPRINT.LE.0)   RETURN
      WRITE(N6,2001) E1,TIME,JSIDE,KWEPTY,KSA,KUA,JAM,RNG,KTGTTY,
     1        JPT,KU,KSB,KPB
      IF (E1.LT.200.)   RETURN
      IPRINT = 2
      GO TO 9001
C***  BLU OR RED WEAPON
 24   KWEPTY = KBST(KSA)
      NAWT1 = NABSS(1,KSA)
      NAWT2 = NABSS(2,KSA)
      NAWU1 = NAMBU(KUA,1)
      NAWU2 = NAMBU(KUA,2)
      KWUTYP= KBUK(2,KUA)
      KADD = 70
      KWKILL = KSA
      KUKILL = KUA
      GO TO 28
 26   KWEPTY = KRST(KSA)
      NAWT1 = NARSS(1,KSA)
      NAWT2 = NARSS(2,KSA)
      NAWU1 = NAMRU(KSA,1)
      NAWU2 = NAMRU(KSA,2)
      KWUTYP= KRUK(2,KUA)
      KADD = 90
      KWKILL = NBSS + KSA
      KUKILL = KBU + KUA
 28   KWEP = MOD(KWEPTY,100)
      E1  = 28.01
      IF (KWEP.LE.0)   GO TO 22
      KWEPT = KWEPTY / 10000
      KTESTA = KWEPT / 100
      KWEPTR = MOD(KWEPTY/100,100)
      DO 32 I=2,9
      KTEST = KADD + I
      IF (KTEST.EQ.KTESTA)  GO TO 34
 32   CONTINUE
      E1  = 32.01
      GO TO 22
 34   KTRANW = I - 1
C***  BLU OR RED TARGET
      GO TO (36,46), JSIDE
```

```
C***  RED TARGET                                                        KILX1600
 36    NATU1 = NAMRU(KU,1)                                              KILX1610
       NATU2 = NAMRU(KU,2)                                              KILX1620
       KUNIT = KRUK(2,KU)/10000                                         KILX1630
       GO TO (38,40), JPT                                               KILX1640
C***  SHIP, ACFT, SUB   -  TARGET  -  RED                               KILX1650
 38    KULOST = KBU + KU                                                KILX1660
       KTARG = KU                                                       KILX1670
       KTGTTY = KRUK(2,KU)                                              KILX1680
       VALTGT = RPFL(4,KU)                                              KILX1690
       KTEST = KRPT(KPB)                                                KILX1700
       F1  = 38.03                                                      KILX1710
       IF (KTEST.NE.KTGTTY)    GO TO 22                                 KILX1720
       KTGTT  = KTGTTY / 10000                                          KILX1730
       KTG = KPB                                                        KILX1740
       ELN = RPX1(1,KTG)                                                KILX1750
       SPN = RPX1(2,KTG)                                                KILX1760
       HGT = RPX1(3,KTG)                                                KILX1770
       VULO = RPX1(11,KTG)                                              KILX1780
       VUL1 = RPX1(12,KTG)                                              KILX1790
       GO TO 60                                                         KILX1800
C***  SSM  &  ASM    -    TARGET   -    RED                             KILX1810
 40    KTGTTY = KRST(KSB)                                               KILX1820
       NATW1 = NARSS(1,KSB)                                             KILX1830
       NATW2 = NARSS(2,KSB)                                             KILX1840
       KTARG = KSB                                                      KILX1850
 42    CONTINUE                                                         KILX1860
       VALTGT=0.                                                        KILX1870
       KTGTT = KTGTTY / 10000                                           KILX1880
       KTG = MOD(KTGTTY,100)                                            KILX1890
       ELN = DATMSL(16,KTG)                                             KILX1900
       SPN = DATMSL(17,KTG)                                             KILX1910
       HGT = DATMSL(18,KTG)                                             KILX1920
       RVUL  = DATMSL(20,KTG)                                           KILX1930
       IF (RVUL.GT..001)    GO TO 44                                    KILX1940
       RVUL  = .5*(VULFM*ELN*SPN*HGT)**.333333                          KILX1950
 44    GO TO 60                                                         KILX1960
C***  BLU TARGET                                                        KILX1970
 46    NATU1 = NAMBU(KU,1)                                              KILX1980
       NATU2 = NAMBU(KU,2)                                              KILX1990
       KUNIT = KBUK(2,KU)/10000                                         KILX2000
       GO TO (48,50), JPT                                               KILX2010
C***  SHIP, ACFT, SUB  -  TARGET  -  BLU                                KILX2020
 48    KULOST = KU                                                      KILX2030
       KTARG = KU                                                       KILX2040
       KTGTTY = KBUK(2,KU)                                              KILX2050
       VALTGT = BRFL(4,KU)                                              KILX2060
       KTEST  = KBPT(KPB)                                               KILX2070
       F1  = 48.03                                                      KILX2080
       IF (KTEST.NE.KTGTTY)    GO TO 22                                 KILX2090
       KTGTT = KTGTTY / 10000                                           KILX2100
       KTG = KPB                                                        KILX2110
       ELN = BPX1(1,KTG)                                                KILX2120
       SPN = BPX1(2,KTG)                                                KILX2130
       HGT = BPX1(3,KTG)                                                KILX2140
```

```
      VULO = BPX1(11,KTG)
      VUL1 = BPX1(12,KTG)
      GO TO 60
C***  SSM  &  ASM    -    TARGET   -   BLU
 50   KTGTTY = KBST(KSB)
      NATW1 = NABSS(1,KSB)
      NATW2 = NABSS(2,KSB)
      KTARG = KSB
      GO TO 42
C***
C***  BRANCH FOR WEAPON TYPES
 60   RWQ = -2.
      SIQ = -2.
      BI  = -2.
      SIGA= -2.
      SIGC = -2.
      CCAP = -2.
      PCAP = -2.
      PKSS = -2.
      GO TO (200,200,200,500,800,700,800,900), KTRANW
C***  TORPEDOES, ASWP, ASROC, FRAS   -    WEAPON KILL
 200  RMX = ZZSYS( 1,KWEP)
      VEL = ZZSYS( 2,KWEP)
      WWT = ZZSYS( 3,KWEP)
      REL = ZZSYS( 4,KWEP)
      SAL = ZZSYS( 5,KWEP)
      SIG = ZZSYS( 7,KWEP) * .849
      BIA = ZZSYS( 8,KWEP)
      RL1 = ZZSYS( 9,KWEP)
      PKE = ZZSYS(10,KWEP)
      WHT = ZZSYS(14,KWEP)
      PHIT= ZZSYS(15,KWEP)
      IF (SAL.LT. .01)   SAL=1.
      IF (RNG.LT..0)   RNG = .8*RMX
      IF (PKE.GT. 1.E-6)   GO TO 202
 202  RVUL = .5*(VULFS*FLN*SPN*HGT)**.333333
C***  BRANCH FOR WARHEAD TYPE
      IF (WHT.GT. .01)   GO TO 212
C***  HE WARHEAD V SHIPS
      IF (PHIT.GT. .0001)   GO TO 206
 203  BI  = .849*BIA*RNG*6.080
      SIQ = SIG**2 + BI**2
 204  RWQ = (RVUL+RL1)**2
      PHIT = 1.
 205  IF (SIQ.LT. .001)   GO TO 206
      PHIT = 1. - EXP(-.5 * RWQ/SIQ)
 206  CALL  KILLHE(VULO,KULOST,VALTGT,WWT,REL,SAL,PHIT,KUKILL,
     1              KWKILL, PK)
      GO TO 9000
C
C***  NUCLEAR WARHEAD VS SHIP
 212  RWQ = (RVUL+RL1)**2
      IF (PHIT.LT. .0001)   GO TO 214
 213  SIQ = -.5*RWQ/ALOG(1.-PHIT)
      GO TO 216
```

```
214   BI   = .849*BIA*RNG*6.080                                          KILX2700
      SIQ = SIG**2 + BI**2                                               KILX2710
C***  ADJUST FOR  RL1.GT.0  (CAPTURE RADIUS FOR HOMING TORPEDOES,ETC.)   KILX2720
215   IF (RL1.LT. 1.)    GO TO 216                                       KILX2730
      IF (RVUL.LT. 10.)    RVUL=10.                                      KILX2740
      SIQ = (SIQ*RVUL**2) / (RVUL+RL1)**2                                KILX2750
216   CALL  KILLNW(VUL0,VUL1,KULOST,VALTGT,WWT,REL,SAL,SIQ,              KILX2760
     1              KUKILL,KWKILL,PKSS,PK)                               KILX2770
      GO TO 9000                                                         KILX2780
C***                                                                     KILX2790
C***  SURFACE TO AIR GUNS     -    WEAPON KILL                           KILX2800
500   GO TO 700                                                          KILX2810
C***                                                                     KILX2820
C***  SURFACE TO AIR MISSILES    -    WEAPON KILL                        KILX2830
      DATA  PHARDA,PHARDM/ 10.,  5./                                     KILX2840
700   RMX = SASYS( 1,KWEP)                                               KILX2850
      TOF = 3600.* TOFG                                                  KILX2860
      ANGLE = 10.                                                        KILX2870
      CALL TLU23(RT,L,M,N,KWEP,TOF,ANGLE,RTF,INDY,INDX,IAY,INX)          KILX2880
      VEL = SASYS( 2,KWEP)                                               KILX2890
      WWT = SASYS( 3,KWEP)                                               KILX2900
      REL = SASYS( 4,KWEP)                                               KILX2910
      SAL = SASYS( 5,KWEP)                                               KILX2920
      SIG = SASYS( 7,KWEP) * .849                                        KILX2930
      BIA = SASYS( 8,KWEP)                                               KILX2940
      RL1 = SASYS( 9,KWEP)                                               KILX2950
      PKF = SASYS(10,KWEP)                                               KILX2960
      WHT = SASYS(14,KWEP)                                               KILX2970
      PHIT= SASYS(15,KWEP)                                               KILX2980
      RWQ = -2.                                                          KILX2990
      SIQ = -2.                                                          KILX3000
      BI  = -2.                                                          KILX3010
      IF (SAL.LT. .01)    SAL=1.                                         KILX3020
C***  IS TARGET AN AIRCRAFT OR A MISSILE                                 KILX3030
      KTEST = KTGTT / 100                                                KILX3040
      KAM = 2                                                            KILX3050
      IF (KTEST.EQ.62.OR.KTEST.EQ.82)    KAM=1                           KILX3060
      PK  = PKF                                                          KILX3070
      IF (PKF.GT. 1.E-6)   GO TO 750                                     KILX3080
      IF (PHIT.GT.1.E-6)   GO TO 740                                     KILX3090
      PHARD = PHARDM                                                     KILX3100
      IF (KAM.EQ.2)    GO TO 708                                         KILX3110
      PHARD = PHARDA                                                     KILX3120
      RVUL = .5*(VULFA*ELN*SPN*HGT)**.333333                             KILX3130
708   KTEST = WHT + .01                                                  KILX3140
      IF (KTEST.NE.2)    GO TO 710                                       KILX3150
      IF (RL1.LT. .01)    RL1= 5.                                        KILX3160
      GO TO 712                                                          KILX3170
C***  HE WARHEAD                                                         KILX3180
710   IF (RL1.LT. .01)    RL1=.23*WWT/SQRT(PHARD)                        KILX3190
712   RWQ = (RVUL+RL1)**2                                                KILX3200
      BI  = BIA                                                          KILX3210
      SIQ = SIG**2 + BI**2                                               KILX3220
      PHIT = 1.                                                          KILX3230
      IF (SIQ.LT. .001)    GO TO 740                                     KILX3240
      PHIT = 1. - EXP(-.5*RWQ/SIQ)
      GO TO 740
C***
 74C  NSAL = SAL + .01
      PHIT = 1. - (1.-PHIT*REL)**NSAL
      PK   = PHIT
C***  KILL RECORD
 75C  XRN = URN(DUMMY)
      PRNG = 1.
      IF (RTF.LE..75*RMX .AND.RTF.GE..25*RMX)  GO TO 7501
      PRNG = 0.8
      IF (RTF.LE..0 .OR. RTF.GE.RMX)   GO TO 7501
      X   = RTF / RMX
      PRNG = -3.2*(X-.25)**2 + 1.0
      IF (X.GE..75)   PRNG = -3.2*(X-.75)**2 + 1.0
 75C1 PK = PK * PRNG
      KSTAA = 1
      IF (XRN.LT.PK )   KSTAA = 0
      IF (KSTAA.GE.1)   GO TO 9010
      VUKILL(KUKILL) = VUKILL(KUKILL) + VALTGT
      VWKILL(KWKILL) = VWKILL(KWKILL) + VALTGT
      GO TO (752,754), KAM
C***  AIRCRAFT TARGET ONLY
 752  VULOST(KULOST) = VALTGT + VULOST(KULOST)
      KSTATE(KULOST) = 0
      GO TO 9012
C***  MISSILE TARGET ONLY
 754  DATMSL(LATMSL,KTG) = DATMSL(LATMSL,KTG) + 1.
      GO TO 9012
C****
C***  SSM  &  ASM   -   WEAPON KILL
 80C  RWQ  = -2.
      SIQ  = -2.
      BI   = -2.
      SIGA = -2.
      SIGC = -2.
      CCAP = -2.
      PCAP = -2.
      RMX  = DATMSL( 1,KWEP)
      VEL  = DATMSL( 2,KWEP)
      WWT  = DATMSL( 3,KWEP)
      REL  = DATMSL( 4,KWEP)
      SAL  = DATMSL( 5,KWEP)
      SIG  = DATMSL( 7,KWEP) * .849
      BIA  = DATMSL( 8,KWEP)
      CODEM= DATMSL( 9,KWEP)
      SIGJ = DATMSL(1C,KWEP) * .849
      CTERM= DATMSL(11,KWEP)
      CMID = DATMSL(12,KWEP)
      TRAJ = DATMSL(13,KWEP)
      WHT  = DATMSL(14,KWEP)
      PHIT = DATMSL(15,KWEP)
C***
      IF (SAL.LT. .01)   SAL=1.
      IF (RNG.LT. .0)   RNG = .8*RMX
      RL1 = .0                                                          KILX3800
      RVUL = .5*(VULFS*ELN*SPN*HGT)**.333333                            KILX3810
      IF (JAM.GE.1)   SIG = SIGJ                                        KILX3820
      KWEP3 = MOD(KWEPT/10,10)                                          KILX3830
      KWEP4 = MOD(KWEPT,10)                                             KILX3840
C***  CASE NO.  1                                                       KILX3850
      IF (PHIT.LT. .0001)   GO TO 804                                   KILX3860
      IF (WHT.LT. .01)   GO TO 206                                      KILX3870
      GO TO 212                                                         KILX3880
C***  ANALYZE GUIDANCE                                                  KILX3890
 804  KGTERM= GTERM + .1                                                KILX3900
      KGTXX = MOD(KGTERM,100)                                           KILX3910
      KGMID = GMID + .1                                                 KILX3920
      KGMXX = MOD(KGMID ,100)                                           KILX3930
      KODM = (CODEM*100.) + .1                                          KILX3940
      IF (KGTXX.NE.0 .OR. KGMXX.NE.0)    GO TO 806                      KILX3950
C***  NO GUIDANCE TYPE SPECIFIED.  USE CEP,BIAS,RL1 INPUTS              KILX3960
      IF (WHT.LT. .01)   GO TO 203                                      KILX3970
      GO TO 214                                                         KILX3980
 806  IF (KODM.LE.0)   GO TO 808                                        KILX3990
C***  MIDCOURSE GUIDANCE SETUP                                          KILX4000
      IG  = KODM / 10000                                                KILX4010
      IE  = MOD(KODM/1000,10)                                           KILX4020
      IL  = MOD(KODM/100,10)                                            KILX4030
      PLP = MOD(KODM/10,10) * .1                                        KILX4040
      PLV = (KODM/10) * .1                                              KILX4050
      E1  = 806.06                                                      KILX4060
      IF (VEL.LE. 1.)   GO TO 22                                        KILX4070
      TH  = RNG / VEL                                                   KILX4080
      PSI = 1.                                                          KILX4090
      E1  = 806.08                                                      KILX4100
      IF (IG*IE*IL.LE.0)   GO TO 22                                     KILX4110
      IF (IG.GT.3.OR.IE.GT.2.OR.IL.GT.3)   GO TO 22                     KILX4120
      CALL  GUID(TH,PSI,VEL,IG,IE,IL,PLP,PLV,SXT,SYT,A)                 KILX4130
      SIGA = .849 * A * 6080.                                           KILX4140
C***  HOMING RADAR SETUP                                                KILX4150
 808  IF (KWEPTR.LE.0)   GO TO 810                                      KILX4160
      SFOV = DATHOM(28,KWEPTR) * .0174533                               KILX4170
      IF (SFOV.GE. 3. .OR. SFOV.LT. .0)   SFOV = 3.                     KILX4180
      BETALF = DATHOM(1,KWEPTR)                                         KILX4190
      IF (JAM.GE.1)   BETALF=DATHOM(2,KWEPTR)                           KILX4200
      IF (BETALF.GT.RNG)   BETALF=RNG                                   KILX4210
      DCAP = BETALF * TAN(.5*SFOV) * 6080.                              KILX4220
 810  IF (KGTXX/10.EQ.4)   GO TO 820                                    KILX4230
C***                                                                    KILX4240
C***  NO TERMINAL HOMING RADAR            *********                     KILX4250
      IF (KGMXX.LT.51 .OR. KGMXX.GT.53)   GO TO 814                     KILX4260
C***  CASE 4.1                                                          KILX4270
C***  AUTOPILOT, DOPPLER, OR INERTIAL + TRACK COMMAND                   KILX4280
      E1  = 810.02                                                      KILX4290
      IF (KODM.LE.0)   GO TO 22                                         KILX4300
      BI  = .849*BIA*RNG*6.080                                          KILX4310
      IF (BI.GT.SIGA)   BI=SIGA                                         KILX4320
      SIO = SIG**2 + BI**2                                              KILX4330
 812  IF (WHT.GT. .01)   GO TO 216                                      KILX4340
```

```
      GO TO 204
  814 IF (KGMXX.NE.54)   GO TO 818
C***  CASE NO.  4.2
C***  INERTIAL ONLY (LAUNCH & FORGET)  -   NO TERMINAL
      E1  = 812.01
      IF (KODM.LE. 0)    GO TO 22
      SIQ = SIG**2 + SIGA**2
      GO TO 812
C***  CASE NO.  4.3
C***  ANY MIDCOURSE + VIDEO  -  NO TERMINAL
  818 E1  = 818.00
      IF (KGMXX.NE.55)   GO TO 22
  819 SIQ = SIG**2
      GO TO 812
C***
C***  WITH TERMINAL HOMING RADAR         ***********
  820 E1  = 820.00
      IF (KWEPTR.LE.0)   GO TO 22
      IF (KGMXX.NE.0)    GO TO 822
C***  CASE NO.  5.
C***  TERMINAL, NO MIDCOURSE
      GO TO 819
  822 E1  = 822.00
      IF (KGMXX.LT.51.OR.KGMXX.GT.53)   GO TO 830
      IF (KODM.LE.0)   GO TO 22
C***  CASE NO.  6.1
C***  TERMINAL + MIDCOURSE (AUTO,DOPP,INERT + TC)
      R    = RNG - BETALF
      PCAP = 1.
      IF (R.LT. 5.0001)   GO TO 824
      TH   = R / VEL
      CALL  GUID(TH,PSI,VEL,IG,IE,IL,PLP,PLV,SXT,SYT,A)
      SIGC = .849 * A * 6080.
      IF (SIGC.LE. 10.001)   GO TO 824
      PCAP = 1. - EXP(-.5*(DCAP/SIGC)**2)
  824 PHIT = 1.
      IF (SIG.LT. 1.)   GO TO 826
      PHIT = 1. - EXP(-.5*(RVUL/SIG)**2)
  826 PHIT = PHIT * PCAP
      IF (WHT.LT. .01)   GO TO 206
      IF (PHIT.LT. .0002)   GO TO 9000
      GO TO 212
C***  CASE NO   6.2
C***  ANY + VIDEO + TERMINAL
  830 E1  = 830.00
      IF (KGMXX.NE.55)  GO TO 22
      GO TO 819
C***
C***  AIR TO AIR MISSILES & GUNS   -   WEAPON KILL
  900 REL = ZZSYS( 4,KWEP)
      RMX = ZZSYS( 1,KWEP)
      VEL = ZZSYS( 2,KWEP)
      WWT = ZZSYS( 3,KWEP)
      SAL = ZZSYS( 5,KWEP)
      SIG = ZZSYS( 7,KWEP) * .849
      BIA = ZZSYS( 8,KWEP)                                              KILX4900
      RL1 = ZZSYS( 9,KWEP)                                              KILX4910
      PKE = ZZSYS(10,KWEP)                                              KILX4920
      WHT = ZZSYS(14,KWEP)                                              KILX4930
      PHIT= ZZSYS(15,KWEP)                                              KILX4940
      RWQ = -2.                                                         KILX4950
      SIQ = -2.                                                         KILX4960
      PI  = -2.                                                         KILX4970
      PK  = -2.                                                         KILX4980
      IF (RNG.GT..0)    GO TO 902                                       KILX4990
      RNG = .8 * RMX                                                    KILX5000
      RTF = .6 * RMX                                                    KILX5010
      TOFG= RTF / VEL                                                   KILX5020
      GO TO 904                                                         KILX5030
  902 RTF = VEL * TOFG                                                  KILX5040
  904 CONTINUE                                                          KILX5050
      IF (SAL.LT. .01)   SAL= 1.                                        KILX5060
C***  IS TARGET AN AIRCRAFT OR A MISSILE                                 KILX5070
      KTEST = KTGTT/100                                                 KILX5080
      KAM = 2                                                           KILX5090
      IF (KTEST.EQ.62.OR.KTEST.EQ.82)    KAM=1                          KILX5100
      PK  = PKE                                                         KILX5110
      IF (PKE.GT. 1.E-6)   GO TO 750                                    KILX5120
      IF(PHIT.GT.1.E-6)   GO TO 740                                     KILX5130
      PK  =  .12345                                                     KILX5140
      GO TO 750                                                         KILX5150
C***                                                                    KILX5160
C***  PRINT OPTIONS                                                     KILX5170
 9000 KSTAA = KSTATE(KULOST)                                            KILX5180
      IF (IPRINT.LE.0)   RETURN                                         KILX5190
      NAMELIST/WEAPA/TIME,JSIDE,KWEAP,KWEPT,KWUNI,KWUTYP,PHIT,          KILX5200
     1               PCAP,VWK,VUK                                       KILX5210
      NAMELIST/WEAPB/RWQ,RVUL,RL1,PHIT,SIQ,SIG,BI,SIGA,                 KILX5220
     1          SIGC,DCAP,PCAP                                          KILX5230
      NAMELIST/TARGA/TIME,JSIDE,KTARG,KTGTT,PKSS,CUM,PKLST,VULST,KSTAT  KILX5240
      NAMELIST/TARGB/VALTGT,VULO,VUL1,RVUL                              KILX5250
      NAMELIST/TARGC/TIME,JSIDE,KTARG,KTGTT,KTGTU,KUNIT                 KILX5260
 9001 CUM    = CUMWT(KULOST)                                            KILX5270
      PKLST = PKLAST(KULOST)                                            KILX5280
      VULST  = VULOST(KULOST)                                           KILX5290
      KSTAT = KSTATE(KULOST)                                            KILX5300
      VUK    = VUKILL(KUKILL)                                           KILX5310
      VWK    = VWKILL(KWKILL)                                           KILX5320
 9002 WRITE(N6,WEAPA)                                                   KILX5330
      IF (IPRINT.GE.2)   WRITE(N6,WEAPB)                                KILX5340
      K    = KTGTT/100                                                  KILX5350
      IF (K.GE.61.AND.K.LE.63)   GO TO 9004                             KILX5360
      IF (K.GE.81.AND.K.LE.83)   GO TO 9004                             KILX5370
C***  NOT A SHIP, AIRPLANE, OR SUB                                      KILX5380
      WRITE(N6,TARGC)                                                   KILX5390
      RETURN                                                            KILX5400
 9004 WRITE(N6,TARGA)                                                   KILX5410
      IF (IPRINT.GE.2)   WRITE(N6,TARGB)                                KILX5420
      RETURN                                                            KILX5430
C***                                                                    KILX5440
```

```
C***  PRINT OPTIONS FOR SAMS & GUNS                                    KILX5450
C***  NO KILL                                                          KILX5460
 9010  IF (IPRINT.LE.0)  RETURN                                        KILX5470
       NAMELIST/WEAPC/TIME,JSIDE,KWEAP,KWEPT,KWUNI,KWUTYP,XRN,PK       KILX5480
       NAMELIST/WEAPD/TIME,JSIDE,KWEAP,KWEPT,KWUNI,KWUTYP,XRN,         KILX5490
     1              PK,VWK,VUK                                         KILX5500
       NAMELIST/WEAPE/RWQ,RVUL,RL1,PHIT,SIQ,SIG,BI                     KILX5510
       NAMELIST/TARGD/TIME,JSIDE,KTARG,KTGTT,KTGTU,KUNIT,VULST,KSTAT   KILX5520
       NAMELIST/TARGE/TIME,JSIDE,KTARG,KTGTT,KTGTU,KUNIT,NULST,KSTAT   KILX5530
       WRITE(N6,WEAPC)                                                 KILX5540
       WRITE(N6,TARGC)                                                 KILX5550
       RETURN                                                          KILX5560
C***  KILL                                                             KILX5570
 9012  IF (IPRINT.LE.0)   RETURN                                       KILX5580
       VUK = VUKILL(KUKILL)                                            KILX5590
       VWK = VWKILL(KWKILL)                                            KILX5600
       WRITE(N6,WEAPD)                                                 KILX5610
       IF (IPRINT.GE.2)   WRITE(N6,WEAPE)                              KILX5620
       GO TO (9016,9018), KAM                                          KILX5630
C***  AIRCRAFT TARGET                                                  KILX5640
 9016  VULST = VULOST(KULOST)                                          KILX5650
       KSTAT = KSTAA                                                   KILX5660
       WRITE(N6,TARGD)                                                 KILX5670
       RETURN                                                          KILX5680
C***  MISSILE TARGET                                                   KILX5690
 9018  NULST = DATMSL(LATMSL,KTG) + .1                                 KILX5700
       KSTAT = KSTAA                                                   KILX5710
       WRITE (N6,TARGE)                                                KILX5720
       RETURN                                                          KILX5730
       END                                                             KILX5740


       SUBROUTINE KILLHE(VULO,KULOST,VALTGT,WWT,REL,SAL,PHIT,KUKILL,   KILH0010
     1              KWKILL,PK)                                         KILH0020
C  PGM=NXX(NEM).  L.D.G.       12-5-73   FORTRAN IV          EBCD      KILH0030
C  MOCS.KT=100,549,550.CVAPK=330,670.CVCGVUL=220,230   VER.3  12-30-74 KILH0040
C   TO KILL SHIPS WITH  HE  OR  NUCLEAR WARHEADS                       KILH0050
C     AND UPDATE RECORDS                                               KILH0060
C  PK     = CUM PROB OF KILL                                           KILH0070
C  VULC   =  SHIP VULNERABILITY CODE (VN)  =                           KILH0080
C  KULOST=  SHIP UNIT HIT                                              KILH0090
C  VALTGT=  SHIP UNIT VALUE                                            KILH0100
C  WWT    =  WARHEAD WT, LBS OF HE,  KT FOR NUCLEAR                    KILH0110
C  REL    =  RELIABILITY                                               KILH0120
C  SAL    =  SALVO SIZE                                                KILH0130
C  PHIT   =  PROB OF HIT                                               KILH0140
C  KUKILL=  PLATFORM UNIT WHOSE WEAPON DID THE KILL                    KILH0150
C  KWKILL=  WEAPON TYPE POINTER WHICH DID THE KILL                     KILH0160
CKILLF                                                                 KILH0170
       COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,       KILH0180
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULOST(100),      KILH0190
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                       KILH0200
C**   BSHP, CSHP = CURVE FIT FOR HE KILL BY SHIP TYPE                  KILH0210
       DIMENSION  BSHP(4), CSHP(4)                                     KILH0220
```

```
      DATA   BSHP/5625.,4388.,1607., 910./                              KILH0230
      DATA   CSHP/2.177,1.5216,2.174,1.721/                             KILH0240
C***  HE WARHEAD KILL                                                   KILH0250
 1    CUM = CUMWT(KULOST) + WWT*REL*SAL*PHIT                            KILH0260
      CUMWT(KULOST) = CUM                                               KILH0270
      IV   = VULO + .01                                                 KILH0280
      PK   = 1. - EXP(-(CUM/BSHP(IV))**CSHP(IV))                        KILH0290
      XRN  = URN(DUMMY)                                                 KILH0300
      K    = KULOST                                                     KILH0310
      CALL   STATE(KSTATE(K),PKLAST(K),PPROD(K), PK,XRN)                KILH0320
      IF (VULO.GT. 1.1)    GO TO 10                                     KILH0330
      IF (KSTATE(K).LE.0.AND.PK.LT..9)  KSTATE(K)=1                     KILH0340
 10   VAL  = PKLAST(K) * VALTGT                                         KILH0350
      DVAL  = VAL - VULOST(K)                                           KILH0360
      VULOST(K) = VAL                                                   KILH0370
      VUKILL(KUKILL) = VUKILL(KUKILL) + DVAL                            KILH0380
      VWKILL(KWKILL) = VWKILL(KWKILL) + DVAL                            KILH0390
      RETURN                                                            KILH0400
C******                                                                 KILH0410
      ENTRY  KILLNW(VULO,VUL1,KULOST,VALTGT,WWT,REL,SAL,SIQ,KUKILL,     KILH0420
     1             KWKILL,PKSS,PK)                                      KILH0430
C*****$*                                                                KILH0440
C   TO KILL SHIPS WITH NUCLEAR WARHEADS                                 KILH0450
C   SIQ   = SIGMA SQUARED, ACCURACY IN FEET SQUARED.                    KILH0460
C   PKSS  = SINGLE SHOT PROB OF KILL                                    KILH0470
C   PK    = CUM PROB OF KILL                                            KILH0480
 20   PK   = .0                                                         KILH0490
      IVN = (100.*VUL1) + .1                                            KILH0500
      IF (IVN.LE.0)    RETURN                                           KILH0510
      XX   = IVN / 1000                                                 KILH0520
      Y    = MOD(IVN/100,10)                                            KILH0530
      F    = MOD(IVN/10,10)                                             KILH0540
      B    = MOD(IVN,10)                                                KILH0550
      CALL   WRAD(XX,Y,F,B,WWT/1000.,RNUC)                              KILH0560
      IF (RNUC.LE..001)    RETURN                                       KILH0570
      RWQ = (RNUC*6080.)**2                                             KILH0580
      PKSS = 1. - EXP(-.5/(.04+(SIQ/RWQ)))                              KILH0590
      K    = KULOST                                                     KILH0600
      NSAL  = SAL + .1                                                  KILH0610
      DO 26 I= 1,NSAL                                                   KILH0620
      PK    = 1. - (1.-PKLAST(K)) * (1.-PKSS*REL)                       KILH0630
      XRN   = URN(DUMMY)                                                KILH0640
      CALL   STATE(KSTATE(K),PKLAST(K),PPROD(K), PK,XRN)                KILH0650
C**   NO COMPLETE A/C CARRIER KILL                                      KILH0660
      IF (VULO.GT. 1.1)    GO TO 24                                     KILH0670
      IF (KSTATE(K).LE.0.AND.PK.LT..9)  KSTATE(K)=1                     KILH0680
 24   VAL  = PKLAST(K) * VALTGT                                         KILH0690
      DVAL = VAL - VULOST(K)                                            KILH0700
      VULOST(K) = VAL                                                   KILH0710
      VUKILL(KUKILL) = VUKILL(KUKILL)+ DVAL                             KILH0720
      VWKILL(KWKILL) = VWKILL(KWKILL) + DVAL                            KILH0730
 26   CONTINUE                                                          KILH0740
C**   ADJUST NUCLEAR KILL TO EQUIV HE WEIGHT                            KILH0750
      IV   = VULO + .01                                                 KILH0760
      CUMWT(KULOST) = BSHP(IV) * ((-ALOG(1.-PK))**(1./CSHP(IV)))        KILH0770
      RETURN                                                            KILH0780
      END                                                               KILH0790


      SUBROUTINE DELSEG(INFORM,ISNSEG,ITGSEG,ISEG)                      INFO0010
      COMMON/STRCON/  JPK,IPK                                           INFO0020
      COMMON/INFO/  LENIFO,LENSEG,MAXSEG,NEXSEG,LSTSEG,LZCSEG           INFO0030
      DIMENSION INFORM(1),ISNSEG(1),ITGSEG(1)                           INFO0040
      IST=INFORM(ISEG)                                                  INFO0050
      ISENS=IST/IPK                                                     INFO0060
      ITGT=MOD(IST,IPK)                                                 INFO0070
      IADD=2                                                            INFO0080
      JPOINT=ITGSEG(ITGT)                                               INFO0090
   21 IPOINT=INFORM(ISEG+IADD)                                          INFO0100
      LSEG=IPOINT/IPK                                                   INFO0110
      NSEG=MOD(IPOINT,IPK)                                              INFO0120
      IF(LSEG.EQ.0) GO TO 211                                           INFO0130
      INFORM(LSEG+IADD)=INFORM(LSEG+IADD)/IPK*IPK+NSEG                  INFO0140
      IF(NSEG.EQ.0) GO TO 215                                           INFO0150
  210 INFORM(NSEG+IADD)=MOD(INFORM(NSEG+IADD),IPK)+LSEG*IPK             INFO0160
      GO TO 218                                                         INFO0170
  211 IF(NSEC.EQ.0) GO TO 212                                           INFO0180
      JPOINT=MOD(JPOINT,IPK)+NSEG*IPK                                   INFO0190
      GO TO 210                                                         INFO0200
  212 JPOINT=C                                                          INFO0210
      GO TO 218                                                         INFO0220
  215 JPOINT=JPOINT/IPK*IPK+LSEG                                        INFO0230
  218 IF(IADD.EQ.1) GO TO 219                                           INFO0240
      ITGSEG(ITGT)=JPOINT                                               INFO0250
      IADD=1                                                            INFO0260
      JPOINT=ISNSEG(ISENS)                                              INFO0270
      GO TO 21                                                          INFO0280
  219 ISNSFG(ISENS)=JPOINT                                              INFO0290
      INFORM(ISEG)=NEXSEG                                               INFO0300
      NEXSEG=ISFG                                                       INFO0310
      RETURN                                                            INFO0320
      END                                                               INFO0330


      SUBROUTINE  STATF(KSTATE,PKLAST,PPROD, PK, URN)                   STAT0010
C  PGM=NXX(NEM).  L.D.G.  NEW SUBR.                            12-18-74 STAT0020
C  TO ASSIGN KSTATE BY LEVEL OF PK, AND STORE PKLAST, ETC.              STAT0030
C  KSTATE = 5, ALIVE.  ALL SYSTEMS OPERATIONAL                          STAT0040
C         = 4, RADARS. ALL SEARCH RADARS OUT                            STAT0050
C         = 3, SAMS.                                                    STAT0060
C         = 2, GUNS.                                                    STAT0070
C         = 1, NOT USED                                                 STAT0080
C         = 0, SUNK.  REMOVED FROM GAME                                 STAT0090
C***                                                                    STAT0100
 1        IF (KSTATE.LE.0)  GO TO 30                                    STAT0110
          PKLAST = PK                                                   STAT0120
          ISTATE = (5.*PK + 1.00001)                                    STAT0130
          IF (ISTATE.GE.5)  ISTATE=6                                    STAT0140
      KSTATE = 6 - ISTATE
 30   RETURN
      END


      FUNCTION URN(IDUM)
      REAL*8 III,ZZZ,XXX,PPP
      COMMON/URN1/ III,ZZZ,XXX,PPP
    1 ZZZ=III*XXX
      J=ZZZ
      XXX=ZZZ-J+J/PPP
      IF(XXX.GT.1.) GO TO 1
      URN=XXX
      RETURN
      END


      FUNCTION XNRN(IDUM)
      TOTAL=0.
      DO 5 I=1,12
    5 TOTAL=TOTAL+URN(IDUM)
      XNRN=TOTAL-6.
      RETURN
      END


      SUBROUTINE ACMTRM(NWTU,KWU,IWSTAT,TUWXYZ,IWTU,INVWTU,MAXWTU,
     1     KTU,ITSTAT,LSYST,ITTU,MAXTTU,
     2     DATMSL)
      COMMON/ECONST/ VAR(15),IMISC(35)
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP
      DIMENSION DATMSL(24,1)
      DIMENSION IWSTAT(1),ITSTAT(1),IWTU(1),TUWXYZ(8,1)
      DIMENSION LSYST(1),ITTU(1),INVWTU(1)
      DIMENSION JWP(4),WEP(5),JTG(5),TGT(5)
      DATA JWP,JTG,WEP,TGT/9*0,      10*0./
      NAMELIST/ACMHIT/ TIME,IWPNTU,ITU,IACM,ITGT,KSTATE
      N6=6
      NLPR=IMISC(3)
      ISIDE=IMISC(4)
      IPR=IMISC(5)
      INIT=0
      IF(NWTU.EQ.0) RETURN
      DO 280 ITU=1,NWTU
      IU=ITU+KWU
      IF( IWSTAT(IU).LE.0) GO TO 280
      TINCPT=TUWXYZ(7,ITU)
      IF(TINCPT.GT.TIME) GO TO 280
C
      IWTU1=IWTU(ITU)
      ITGT=IWTU1/100000
      IACM=MOD( IWTU1, 10)
      IF(TINCPT.EQ.-100.) GO TO 250
      IF(ITGT.EQ.0) GO TO 270
      IF(ITSTAT(ITGT).LE.0) GO TO 250
      IWTU2=IWTU(MAXWTU+ITU)
C
      IROW=MOD(IWTU1/1000,100)
      LOOP=1
      LJ=IWTU2/1000000
      IRPD=MOD(LJ,100)
      LJ=LJ/100
      IF(IACM.EQ.2) GO TO 100
      IF(IRPD.GE.IMISC(6)) GO TO 10
C----------   CM HAS TOO LITTLE RADAR DETECTION TO ACQUIRE ITS TGT
      KSTATE=-100
      GO TO 202
   10 JWP(4)=LJ/2
      ICOL=MOD(IWTU1/10,100)
      DATMSL(23,ICOL)=DATMSL(23,ICOL)+1.
      RANGE=MOD(IWTU2/100,10000)
      GO TO 101
  100 IROW=IRPD
      IF(LJ.GT.LOOP) LOOP=LJ
      JWP(4)=0
      RANGE=-10.
  101 JWP(1)=IROW
      JWP(3)=MOD(IWTU2,100)
      WEP(1)=RANGE*.1
      TL=IABS(INVWTU(ITU))/100000
      TC=(TINCPT-TBEGIN)*1000.
      WEP(2)=(TC-TL)*.001
      JTG(2)=ITGT
      JTGT=ITGT-KTU
      IF(JTGT.GT.0) GO TO 102
      JTG(1)=1
      JTG(4)=MOD(LSYST(ITGT),100)
      GO TO 105
  102 ITTU1=ITTU(JTGT)
      JTG(1)=MOD(ITTU1,10)/2
      JTG(2)=MOD(ITTU(MAXTTU+JTGT),100)
      IRP=MOD(ITTU1/1000,100)
      JTG(3)=IRP
      JTG(4)=IRP
  105 CONTINUE
      DO 200 LJ=1,LOOP
      J=LJ
      CALL KILLEX(INIT,ISIDE,JWP,WEP,JTG,TGT,IPR,KSTATE)
      IF(KSTATE.GT.0) ITSTAT(ITGT)=KSTATE*1000+MOD(ITSTAT(ITGT),1000)
      IF(KSTATE.NE.0) GO TO 200
      ITSTAT(ITGT)=-1
      GO TO 201
  200 CONTINUE
  201 IWTU(MAXWTU+ITU)=IWTU2-J*100000000
  202 IF(NLPR.EQ.0) GO TO 250
      IWPNTU=IU
      WRITE(N6,ACMHIT)
```

```
C                                                                       ACMT0820
C  POSSIBLE PRIMARY UNIT KILLED                                         ACMT0830
C  POSSIBLE INTERCEPT UNIT KILLED                                       ACMT0840
C                                                                       ACMT0850
C                                                                       ACMT0860
C  FINAL DISPOSITION INTERCEPT UNIT                                     ACMT0870
C                                                                       ACMT0880
  250 IF( IACM .EQ.4) GO TO 270                                         ACMT0890
      IF( ITGT .EQ.0) GO TO 270                                         ACMT0900
      IWTU( ITU)=MOD(IWTU1,100000)                                      ACMT0910
      INVWTU( ITU)=(IABS( INVWTU( ITU))/100000)*100000                  ACMT0920
      TUWXYZ(8,ITU)=TIME-TSTEP                                          ACMT0930
      TUWXYZ(7,ITU)=TIMAX                                               ACMT0940
      GO TO 280                                                         ACMT0950
  270 IWSTAT( IU)=-10                                                   ACMT0960
  280 CONTINUE                                                          ACMT0970
      RETURN                                                            ACMT0980
      END                                                               ACMT0990


       SUBROUTINE TLU23(ZT,L,M,N,KN,Y,X,Z,INDY,INDX,INY,INX)            TL230010
C  PGM=NU6(CGSM) VER.1, 05-12-71. L.D.GREGORY 3-53300. FORTRAN IV.EBCD  TL230020
C  THREE DIMENSIONAL STORAGE OF 2-WAY TABLES                            TL230030
C   INX AND INY = -1,0,+1 ACCORDING TO IF LT,EQ,GT TABULATED INTERVAL   TL230040
C      DOUBLE INTERPOLATION FROM 2-DIMENSION BORDERED TABLE IN ZT(L,M)  TL230050
C      BORDER VALUES MUST INCREASE.                                     TL230060
C      IF NOT IN TABULATED INTERVAL, USES NEAREST VALUE.                TL230070
C      KN = TABLE NO. IN 3-DIMENSIONAL ARRAY                            TL230080
       DIMENSION ZT( L, M, N)                                           TL230090
       INX=0                                                            TL230100
       INY=0                                                            TL230110
 1     XX=X                                                             TL230120
       YY=Y                                                             TL230130
       IM=(ZT(1,1,KN) + 0.1)                                            TL230140
       IN=MOD(IM,100)                                                   TL230150
       IM=IM/100                                                        TL230160
       YM=ZT(IM,1,KN)                                                   TL230170
       XN=ZT(1,IN,KN)                                                   TL230180
 3     IF (YY-YM) 24, 24, 20                                            TL230190
 20    INY=1                                                            TL230200
       YY=YM                                                            TL230210
       IY=IM                                                            TL230220
 24    IF (XX-XN) 5, 5, 26                                              TL230230
 26    INX=1                                                            TL230240
       XX=XN                                                            TL230250
       IX=IN                                                            TL230260
 5     YS=ZT( 2,1,KN)                                                   TL230270
       XS=ZT(1,2,KN)                                                    TL230280
       IF (YY-YS) 28, 30, 30                                            TL230290
 28    INY=-1                                                           TL230300
       YY=YS                                                            TL230310
       IY=3                                                             TL230320
 30    IF (XX-XS) 32, 34, 34                                            TL230330
 32    INX=-1                                                           TL230340
```

```
      XX=XS                                                             TL230350
      IX=3                                                              TL230360
 34   I=2                                                               TL230370
      CO 10 K=3,IM                                                      TL230380
      I=I+1                                                             TL230390
      IF (ZT(I,1,KN)-YY) 10, 12, 12                                     TL230400
 10   CONTINUE                                                          TL230410
 12   YU=ZT(I,1,KN)                                                     TL230420
      YL=ZT(I-1,1,KN)                                                   TL230430
      IY=I                                                              TL230440
      INDY=I-1                                                          TL230450
      I=2                                                               TL230460
      CO 14 K=3,IN                                                      TL230470
      I=I+1                                                             TL230480
      IF (ZT(1,I,KN)-XX) 14, 16, 16                                     TL230490
 14   CONTINUE                                                          TL230500
 16   XU=ZT(1,I,KN)                                                     TL230510
      XL=ZT(1,I-1,KN)                                                   TL230520
      IX=I                                                              TL230530
      INDX=I-1                                                          TL230540
      AX = (XU-XX)/(XU-XL)                                              TL230550
      AY = (YU-YY)/(YU-YL)                                              TL230560
      ZLL= ZT(IY-1,IX-1,KN)                                             TL230570
      ZLU= ZT(IY-1,IX  ,KN)                                             TL230580
      ZUL= ZT(IY  ,IX-1,KN)                                             TL230590
      ZUU= ZT(IY  ,IX  ,KN)                                             TL230600
      ZL = AX*ZLL+(1.0-AX)*ZLU                                          TL230610
      ZU = AX*ZUL+(1.C-AX)*ZUU                                          TL230620
      Z  = AY*ZL +(1.0-AY)*ZU                                           TL230630
      IF (INY) 40, 42, 41                                               TL230640
 40   INDY=INDY-1                                                       TL230650
      GO TO 42                                                          TL230660
 41   INCY=INDY+1                                                       TL230670
 42   IF (INX) 44, 46, 45                                               TL230680
 44   INDX=INDX-1                                                       TL230690
      GO TO 46                                                          TL230700
 45   INDX=INDX+1                                                       TL230710
 46   RETURN                                                            TL230720
      END                                                               TL230730


      SUBROUTINE WRAD(DAMSUS,POQ,DSKF,GOAB,WWW,RAD)                     WRAD0010
      CIMFNSION    PA(13), PS(13), QA(9), QS(9)                         WRAD0020
      DATA PA /2.3700,1.7805,1.1712,0.5890,0.0250,-0.4694,-0.8881,      WRAD0030
     1         -1.2055,-1.5192,-1.8557,-2.1754,-2.4583,-2.7033/         WRAD0040
      CATA PS /1.8669,1.3202,0.7606,0.2875,-0.1499,-0.5516,-0.9206,     WRAD0050
     1         -1.2166,-1.5192,-1.8557,-2.1754,-2.4583,-2.7033/         WRAD0060
      CATA QA /2.4667,1.9092,1.2136,0.5451,0.0827,-0.3137,-0.7226,      WRAD0070
     1         -1.1678,-10.0000/                                        WRAD0080
      CATA QS /1.9844,1.4382,0.841E,0.2973,-0.1712,-0.5806,-1.0529,     WRAD0090
     1         -1.5497,-10.0000/                                        WRAD0100
      IF (CAMSUS .GE. 75.)              GO TO 90                        WRAD0110
     ADVN=DAMSUS                                                        WRAD0120
     IVN=1                                                              WRAD0130
```

```
      D=5.485                                                           WRAD0140
      IBT=2                                                             WRAD0150
      IF(POQ.LT..01)      GO TO 2                                       WRAD0160
      IVN=2                                                             WRAD0170
      D=2.742                                                           WRAD0180
    2 IF(GOAB.LT..01)      IBT=1                                        WRAD0190
      IF(DSKF.LT..01)      GO TO 60                                     WRAD0200
      A=DSKF*.1                                                         WRAD0210
      Q=A-1.                                                            WRAD0220
      AB=(.02/WWW)**(1./3.)*A                                           WRAD0230
      IF(IVN.EQ.2)      GO TO 5                                         WRAD0240
      R=(AB**2-2.*Q+SQRT((2.*Q-AB**2)**2-4.*Q**2))*.5                   WRAD0250
      GO TO 20                                                          WRAD0260
    5 R=(.02/WWW)**(A*1.58/D)+1.-A                                      WRAD0270
      N=IVN+1                                                           WRAD0280
      DO 10 I=1,50                                                      WRAD0290
      F=(R-1.+A)/AB                                                     WRAD0300
      X=F**N-R                                                          WRAD0310
      IF(ABS(X).LE..0001)      GO TO 20                                 WRAD0320
      XP=N*F**(IVN)/AB                                                  WRAD0330
      R=R-X/XP                                                          WRAD0340
   10 CONTINUE                                                          WRAD0350
   20 ADVN=DAMSUS+D*ALOG(R)                                             WRAD0360
   60  DO 65    I = 2,13                                                WRAD0370
          X    = (I - 1) * 5                                            WRAD0380
          IF (ADVN .LE. X)                   GO TO 70                   WRAD0390
   65  CONTINUE                                                         WRAD0400
       I       =   13                                                   WRAD0410
   70  IF ((I .GT. 9).AND.(IVN .EQ. 2)) GO TO 90                        WRAD0420
       IHI     =   I                                                    WRAD0430
       ILO     =   I - 1                                                WRAD0440
       X       =  (ILO - 1) * 5                                         WRAD0450
       DVN     =   ADVN - X                                             WRAD0460
       GO TO (75,78),IVN                                                WRAD0470
   75  GO TO (76,77),IBT                                                WRAD0480
   76  Y       =   PS(ILO)                                              WRAD0490
       Z       =   PS(IHI)                                              WRAD0500
       GO TO 85                                                         WRAD0510
   77  Y       =   PA(ILO)                                              WRAD0520
       Z       =   PA(IHI)                                              WRAD0530
       GO TO 85                                                         WRAD0540
   78  GO TO (79,80),IBT                                                WRAD0550
   79  Y       =   QS(ILO)                                              WRAD0560
       Z       =   QS(IHI)                                              WRAD0570
       GO TO 85                                                         WRAD0580
   80  Y       =   QA(ILO)                                              WRAD0590
       Z       =   QA(IHI)                                              WRAD0600
   85  SS      =  (Y - Z) * 0.2                                         WRAD0610
       Y       =   Y - (SS * DVN)                                       WRAD0620
       RAD     =   EXP(Y)  * WWW**(1./3.)                               WRAD0630
       GO TO 95                                                         WRAD0640
   90  RAD     =   0.0                                                  WRAD0650
   95  RETURN                                                           WRAD0660
       END                                                              WRAD0670
      SUBROUTINF STATUS(NWTU,INVWTU, KWSEG,IWTU,NEXTU,KWU,ITWSEG,
     1    IWSTAT,MAXWTU,LSYSW,WGC,INTVLW,NWG,NWUG,
     2    KTSEG,ITSTAT,KTU,ITTU,ITTSEG,
     3    INFORM,LBRSYS)
      COMMON/ECONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,PDMIN,
     1    AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)
      EQUIVALENCE (IMISC(20),LRATE)
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP
      CIMENSION ITSTAT(1),INVWTU(1),  KWSEG(1), IWTU(1),ITWSEG(1),
     1    IWSTAT(1),KTSEG(1),INFORM(1),LBRSYS(1),LSYSW(1)
     2,   ITTU(1),ITTSEG(1),WGC(8,6,1),INTVLW(1),NWUG(1)
      CIMENSION KSIDE(2),LPGRP(18,2),KEYSYS(8), KUNIT(50,2)
      DATA KEYSYS/20,12,22,23,33,16,24,43/
      CATA NKEY,KPG/8,3/
      EQUIVALENCE (IMISC(4),ISIDE)
C
      ITIME=(TIME-TBEGIN)*1000.
      NUNIT=KWU+NWTU
      CO 380C IU=1,NUNIT
      ISTAT=IWSTAT(IU)
      ITU=IU-KWU
      IF(ISTAT) 300,380,330
  3C0 ISTAT=IABS(ISTAT)
      IF(ISTAT.EQ.1) GO TO 350
      IF(ISTAT.GT.2000C00000) GO TO 355
      IF(ISTAT.EQ.10) GO TO 340
      IF(ISTAT.LT.100000) GO TO 310
      ISB=MOD(LSYSW(IU)/100,10000)-2
      ICAR=LBRSYS(ISB)/1000000
      IF(IWSTAT(ICAR).GT.0) GO TO 302
  3C1 LBRSYS(ISB)=-LBRSYS(ISB)
      IWSTAT(IU)= -210C000000
      GO TO 355
  302 LINF=ISTAT/1C000C
      JTIMF=MOD(ISTAT,100000)
      LS=LBRSYS(LINF)
      IF(LS.EQ.0) GO TO 301
      IF(LS/1C00000.EQ.0) GO TO 355
      KSB=MOD(LSYSW(ICAR)/100,10000)-2
      IF(LBRSYS(KSB).GT.ITIME*100) GO TO 355
      LBRSYS(KSB)=(ITIME+LRATE)*100
      LBRSYS(LINF)=LS-1000000
      IWSTAT(IU)=-(JTIMF+ITIME)
      GO TO 355
  310 IF(ISTAT.GT.ITIMF) GO TO 355
      IWSTAT(IU)=1
      GO TO 355
  330 IWSTAT(IU)=MOD(ISTAT,1000)
      IF(IU.LF.KWU) GO TO 375
      IF(MOD(IWTU(ITU),10) .FQ. 4) GO TO 380
      ITGT=MOD(IWTU(ITU)/10000C,1000)
      IF(ITSTAT(ITGT).LF.0) GO TO 345
      IF(ITGT.LF.KTU) GO TO 380
      JTU=ITGT-KTU
      JTGT=MOD(ITTU(JTU)/1000C,1000)
      IF(IWSTAT(JTGT))345,345,380
  340 IF(MOD(IWTU(ITU),10) .EQ. 4) GO TO 350
  345 LU=MOD( IWTU(MAXWTU+ITU),100)
      ISB=MOD(LSYSW(LU)/100,10000)
      LINE=MOD(LBRSYS(ISB-2)/100,10000)
      IF(LINE) 350,348,346
  346 LS=LBRSYS(LINE)
      IF(LS.EQ.0) GO TO 350
      LBRSYS(LINE)=LS+1000000
      GO TO 350
  348 IWSTAT(LU)=1
      KROW=IWTU(MAXWTU+ITU)
      KS=KROW/100000000
      KROW=MOD(KROW/1000000,100)
      ISS=ISB-3
      ISA=LSYSW(LU)/1000000
      DO 349 I=ISA,ISS
      LS=LBRSYS(I)
      IF(MOD(LS/100,100) .NE. KROW) GO TO 349
      LBRSYS(I)=MOD(LS,1000000)+KS*1000000+(LS/100000000)*100000000
      GO TO 350
  349 CONTINUE
  350 IWSTAT(IU)=0
      IF(IU.LE.KWU) GO TO 355
  352 IWTU(ITU)=NEXTU
      NEXTU=ITU
      INVWTU(ITU)=0
      GO TO 360
  355 ISEG=KWSEG(IU)/10000
  356 IF(ISEG.EQ.0) GO TO 360
      IF(INFORM(ISEG+7).EQ.0) GO TO 357
      INFORM(ISEG+4)=0
      INFORM(ISEG+5)=0
      INFORM(ISEG+6)=0
      GO TO 358
  357 CALL DELSEG(INFORM,KWSEG,ITTSEG,ISEG)
  358 ISEG=MOD(INFORM(ISEG+1),10000)
      GO TO 356
C
  360 KSEG=ITWSEG(IU)
      ISEG=KSEG/10000
  365 IF(ISEG.EQ.0) GO TO 380
      IWORD=INFORM(ISEG+6)
      IF(IWORD.EQ.0) GO TO 370
      LINE=IWORD/100000
      LBRSYS(LINE)=LBRSYS(LINE)+100000000
  370 CALL DELSEG(INFORM,KTSEG,ITWSEG,ISEG)
      ISEG=MOD(INFORM(ISEG+2),10000)
      GO TO 365
  375 KSTATE=ISTAT/1000
C
      IS=LSYSW(IU)
      ISB=MOD(IS/100,10000)
      IF(ISB.EQ.0) GO TO 380
      MASK=LBRSYS(ISB)
      KW=MOD(MASK,10)
      IF(KW.EQ.2) GO TO 380
      IWSTAT(IU)=ISTAT
  376 IF(KSTATE.EQ.0 .OR. KSTATE.GE.5) GO TO 380
C
 3775 ISA=IS/1000000
      ISB=ISB-3
      DO 379 I=ISA,ISB
      KSYS=MOD(LBRSYS(I),100)
      IF(KSYS.EQ.11) GO TO 378
      IF(KSYS.EQ.21) GO TO 378
      IF(KSTATE.GE.4) GO TO 379
      IF(KSYS.EQ.33) GO TO 378
      IF(KSTATE.GE.3) GO TO 379
      IF(KSYS.EQ.43) GO TO 378
      IF(KSYS.EQ.12) GO TO 378
      IF(KSYS.EQ.20) GO TO 378
      GO TO 379
  378 LBRSYS(I)=0
  379 CONTINUE
  380 CONTINUE
C
C         RENDEVOUS LOGIC
C
      IF(IMISC(ISIDE+12).EQ.0) GO TO 650
      IF(INTVLW(19).NE.0) GO TO 600
      NPG=0
      JKEY=0
      DO 590 IG=1,NWG
      IKEY=JKEY+1
      K=NWUG(IG)
      IUA=K/100
      IUB=IUA+MOD(K,100)-1
      DO 585 K=1,NKEY
      KEY=KEYSYS(K)
      DO 580 IU=IUA,IUB
      IS=LSYSW(IU)/100
      IF(IS.EQ.0) GO TO 580
      ISB=MOD(IS,10000)-3
      ISA=IS/10000
      DO 540 I=ISA,ISB
      LS=LBRSYS(I)
      KSYS=MOD(LS,100)
      IF(KSYS.NE.KEY) GO TO 540
      JKEY=JKEY+1
      KUNIT(JKEY,ISIDE)=IU
      GO TO 580
  540 CONTINUE
  580 CONTINUE
      IF(JKEY.GE.IKEY) GO TO 586
  585 CONTINUE
      INTVLW(IG+18)=-1
      GO TO 590
  586 IF(KEY.GT.KPG) GO TO 587
      NPG=NPG+1
      LPGRP(NPG,ISIDE)=IG                                               STAT1660
  587 INTVLW(IG+18)=KEY*10000+IKEY*100+JKEY                             STAT1670
  590 CONTINUE                                                          STAT1680
      KSIDE(ISIDE)=NPG                                                  STAT1690
C                                                                       STAT1700
  600 NPG=KSIDE(ISIDE)                                                  STAT1710
      IF(NPG) 650,625,605                                               STAT1720
  605 DO 610 I=1,NPG                                                    STAT1730
      IG=LPGRP(I,ISIDE)                                                 STAT1740
      IF(INTVLW(IG+18).NE.1)  GO TO 625                                 STAT1750
  610 CONTINUE                                                          STAT1760
      KSIDE(ISIDE)=-1                                                   STAT1770
      NPG=-1                                                            STAT1780
  625 DO 640 IG=1,NWG                                                   STAT1790
      KEY=INTVLW(IG+18)                                                 STAT1800
      IF(KEY.EQ.1) GO TO 640                                            STAT1810
      IF(NPG.LT.0) GO TO 637                                            STAT1820
      IF(KEY.LT.0) GO TO 640                                            STAT1830
      LEAVE=0                                                           STAT1840
      ILIVE=0                                                           STAT1850
      JKEY=MOD(KEY,100)                                                 STAT1860
      IKEY=MOD(KEY/100,100)                                             STAT1870
      KEY=KEY/10000                                                     STAT1880
      DO 635 K=IKEY,JKEY                                                STAT1890
      IU=KUNIT(K,ISIDE)                                                 STAT1900
      IF(IWSTAT(IU)) 634,635,627                                        STAT1910
  627 IS=LSYSW(IU)/100                                                  STAT1920
      ISB=MOD(IS,10000)-3                                               STAT1930
      ISA=IS/10000                                                      STAT1940
      DO 630 I=ISA,ISB                                                  STAT1950
      LS=LBRSYS(I)                                                      STAT1960
      KSYS=MOD(LS,100)                                                  STAT1970
      IF(KSYS.NE.KEY) GO TO 630                                         STAT1980
      IF(KSYS.EQ.33) LS=LBRSYS(I+1)                                     STAT1990
      IF(MOD(LS/1000000,100).GT.0) GO TO 640                            STAT2000
  630 CONTINUE                                                          STAT2010
      LEAVE=LEAVE+1                                                     STAT2020
  634 ILIVE=ILIVE+1                                                     STAT2030
  635 CONTINUE                                                          STAT2040
      IF(ILIVE.EQ.0) GO TO 637                                          STAT2050
      IF(LEAVE.EQ.0) GO TO 640                                          STAT2060
  637 INTVLW(IG+18)=1                                                   STAT2070
      LEG=IABS(INTVLW(IG))                                              STAT2080
      INTVLW(IG)=10+LEG                                                 STAT2090
      WGC(1,5,IG)=INTVLW(37)                                            STAT2100
      WGC(1,6,IG)=INTVLW(38)                                            STAT2110
  640 CONTINUE                                                          STAT2120
C                                                                       STAT2130
  650 CONTINUE                                                          STAT2140
      RETURN                                                            STAT2150
      END                                                               STAT2160


      SUBROUTINE USTAT(KWU,IWSTAT,ITWSEG,NWTU,KWUK,                     USTA0010
     1       NTTU,INVTTU,ITTU,                                                  USTA0020
     2       INFORM)                                                            USTA0030
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP                 USTA0040
      COMMON /WORK/ KSTAT(500)                                                  USTA0050
      DIMENSION KWUK(4,1)                                                       USTA0060
      DIMENSION IWSTAT(1),ITWSEG(1),INFORM(1),INVTTU(1),ITTU(1)                 USTA0070
      NUNIT=KWU+NWTU                                                            USTA0080
      DO 310 I=1,NUNIT                                                          USTA0090
  310 KSTAT(I)=0                                                                USTA0100
      IF(NTTU.EQ.0) GO TO 350                                                   USTA0110
      DO 340 ITU=1,NTTU                                                         USTA0120
      IF(INVTTU(ITU).EQ.0) GO TO 340                                            USTA0130
      ITGT=MOD(ITTU(ITU)/100000,1000)                                           USTA0140
      IF(ITGT.EQ.0) GO TO 340                                                   USTA0150
      KSTAT(ITGT)=KSTAT(ITGT)+1                                                 USTA0160
  340 CONTINUE                                                                  USTA0170
  350 DO 380 IU=1,NUNIT                                                         USTA0180
      ISTAT=IWSTAT(IU)                                                          USTA0190
      IF(ISTAT.LE.0) GO TO 378                                                  USTA0200
      KSTATE=ISTAT/1000                                                         USTA0210
      NGAGE=KSTAT(IU)                                                           USTA0220
      IF(NGAGE.EQ.0) GO TO 355                                                  USTA0230
      ASSIGN 368 TO ISWT                                                        USTA0240
      ASSIGN 368 TO KSWT                                                        USTA0250
      GO TO 358                                                                 USTA0260
  355 ASSIGN 366 TO ISWT                                                        USTA0270
      ASSIGN 364 TO KSWT                                                        USTA0280
      ISTAT=1                                                                   USTA0290
  358 KSEG=ITWSEG(IU)                                                           USTA0300
      ISEG=KSEG/10000                                                           USTA0310
  360 IF(ISEG.EQ.0) GO TO 370                                                   USTA0320
      IF(INFORM(ISEG+7)) 363,363,362                                            USTA0330
  362 NGAGE=NGAGE+1                                                             USTA0340
      GO TO ISWT,(368,366)                                                      USTA0350
  363 GO TO KSWT,(368,364)                                                      USTA0360
  364 IF(INFORM(ISEG+5)) 368,368,365                                            USTA0370
  365 ISTAT=2                                                                   USTA0380
  366 ASSIGN 368 TO ISWT                                                        USTA0390
      ASSIGN 368 TO KSWT                                                        USTA0400
  368 ISEG=MOD(INFORM(ISEG+2),10000)                                            USTA0410
      GO TO 360                                                                 USTA0420
  370 IF(NGAGE.GT.0) ISTAT=NGAGE+2                                              USTA0430
C                                                                               USTA0440
C  POSSIBLE            UNIT DETECTION LOSS                                      USTA0450
C                                                                               USTA0460
      IWSTAT(IU)=ISTAT                                                          USTA0470
  378 IF(IU.GT.KWU) GO TO 380                                                   USTA0480
      IF(ISTAT) 3784,3786,3782                                                  USTA0490
 3782 JSTATE=KWUK(4,IU)/100                                                     USTA0500
      IF(KSTATE.GT.0) JSTATE=KSTATE                                             USTA0510
      ISTAT=ISTAT+JSTATE*100                                                    USTA0520
      GO TO 3786                                                                USTA0530
 3784 ISTAT=-1                                                                  USTA0540
 3786 KWUK(4,IU)=ISTAT                                                          USTA0550
  380 CONTINUE                                                                  USTA0560
```

```
      RETURN
      END
      SUBROUTINE SNAP(NWU,NWTU,LSYSW,LBRSYS,KWSEG,KTSEG,INFORM,
     1     IWSTAT,IWTU,INVWTU,TUWXYZ,MAXTU,NAMU,NAMS,WXYZ,
     2     TXYZ,TUTXYZ,NTU)
      DIMENSION LSYSW(1),LBRSYS(1),KWSEG(1),KTSEG(1),INFORM(1),
     1     IWSTAT(1),IWTU(1),INVWTU(1),TUWXYZ(8,1)
     2,    TXYZ(50,7),NAMU(50,2),NAMS(2,1)
     3,    WXYZ(50,1),TUTXYZ(8,1)
      COMMON/ECONST/ VAR(15),IMISC(35)
      COMMON/STRCON/  JPK,IPK
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12
      COMMON /ETIME/ TIME,TIMEA,TIMEB,TBEGIN,TIMEND,TIMAX,TSTEP
      EQUIVALENCE (IMISC(4),ISIDE)
      DATA IBLU,IRED/' BLU',' RED'/
      NSIDE=IBLU
      IF(ISIDE.EQ.2) NSIDE=IRED
      IPR=IMISC(7)
      CALL PAGE
      JTIME=(TIME-TBEGIN)*1000.
      IF(IPR.LT.3) GO TO 300
      WRITE(N6,6100) TIME,JTIME,ISIDE
 6100 FORMAT(///' INFORMATION SNAPSHOT, TIME=',F8.4,I7,' SIDE=',I2/ )
      WRITE(N6,6110)
 6110 FORMAT('    SYSTEMS DATA'/)
      DO 110 I=1,NWU
      K=LSYSW(I)
      IF(K.EQ.0) GO TO 110
      ISA=K/1000000
      ISB=MOD(K/100,10000)
      WRITE(N6,6111) I,K,(LBRSYS(L),L=ISA,ISB)
  110 CONTINUE
 6111 FORMAT(1X,I4,3X,10I12/(8X,10I12))
  200 IF(IPR.LT.4) GO TO 300
      WRITE(N6,6100) TIME,JTIME,ISIDE
      WRITE(N6,6210)
 6210 FORMAT('    INFORMATION SEGMENT DATA'/)
      DO 210 I=1,NWU
      K=KWSEG(I)
      IF(K.EQ.0) GO TO 210
      ISEG=K/IPK
      JSEG=ISEG+7
      WRITE(N6,6211) I,K,(INFORM(L),L=ISEG,JSEG)
   21 ISEG=MOD(INFORM(ISEG+1),IPK)
      IF(ISEG.EQ.0) GO TO 210
      JSEG=ISEG+7
      WRITE(N6,6212) (INFORM(L),L=ISEG,JSEG)
      GO TO 21
  210 CONTINUE
 6211 FORMAT(1X,I4,3X,I12,5X,8I12)
 6212 FORMAT(25X,8I12)
      IF(NWTU.EQ.0) GO TO 410
  300 CONTINUE
      WRITE(N6,6100) TIME,JTIME,ISIDE
      WRITE(N6,6310) NSIDE
 6310 FORMAT('    TEMPORARY UNIT DATA FOR',A4/)
      WRITE(N6,6313)
      DO 310 I=1,NWTU
      K=INVWTU(I)
      IF(K.EQ.0) GO TO 310
      ITU=I+NWU
      IWTU1=IWTU(I)
      IAC=MOD(IWTU1,10)
      J=I+MAXTU
      IWTU2=IWTU(J)
      LU=MOD(IWTU2,100)
      IF(IAC.EQ.4) GO TO 304
      NAM1=NAMU(LU,1)
      NAM2=NAMU(LU,2)
      GO TO 305
  304 J=MOD(IWTU1/1000,100)
      NAM1=NAMS(1,J)
      NAM2=NAMS(2,J)
  305 ITGT=IWTU1/100000
      WX=TUWXYZ(1,I)
      WY=TUWXYZ(2,I)
      WZ=TUWXYZ(3,I)
      VX=TUWXYZ(4,I)
      VY=TUWXYZ(5,I)
      IF(ITGT.EQ.0) GO TO 306
      IF(ITGT.GT.NTU) GO TO 307
      DX=TXYZ(ITGT,2)-WX
      DY=TXYZ(ITGT,3)-WY
      GO TO 308
  306 ICAR =MOD(IWTU2/100,100)
      IF(ICAR.EQ.0) ICAR=LU
      DX=WXYZ(ICAR,2)-WX
      DY=WXYZ(ICAR,3)-WY
      GO TO 308
  307 L=ITGT-NTU
      DX=TUTXYZ(1,L)-WX
      DY=TUTXYZ(2,L)-WY
  308 GR=SQRT(DX*DX+DY*DY)
      V=SQRT(VX*VX+VY*VY)
      WRITE(N6,6311) ITU,NAM1,NAM2,IWSTAT(ITU),LU,ITGT,WX,WY,WZ,V,GR,
     1             (TUWXYZ(L,I),L=6,8)
      IF(IPR.LT.2) GO TO 310
      WRITE(N6,6312) IWTU1,IWTU2,K
  310 CONTINUE
 6311 FORMAT(1X,I3,'=',2A4,I3,I6,I6,3X,8G12.5)
 6312 FORMAT(5X,'IWTU=',I12,',',I12,'     INVWTU=',I12)
 6313 FORMAT(1X,'UNIT  NAME  STAT ORIGIN TGTU', 6X,'X,NM',
     1     8X,'Y,NM',8X,'Z,NM',6X,'VXY,KT',7X,'RANGE',7X,'VZ,KT',
     2     5X,'TINCEPT',5X,'TVECTOR'/)
  410 IF(IPR.LT.4) RETURN
C     CALL PAGE
      WRITE(N6,6100) TIME,JTIME,ISIDE
      WRITE(N6,6410)                                                    SNAP1060
 6410 FORMAT('   UNIT STATUS AND TGTSEG CODES'/)                        SNAP1070
      JWU=NWU+NWTU                                                      SNAP1080
      WRITE(N6,6411) (I,IWSTAT(I),KTSEG(I),I=1,JWU)                     SNAP1090
 6411 FORMAT((5(2X,I3,'=',2I10)))                                       SNAP1100
      RETURN                                                            SNAP1110
      END                                                               SNAP1120


      SUBROUTINE   SUMOUT(KPAS)                                         SUM 0010
C  PGM=NXX(NEM)     L.D.G.        12-17-73          FORTRAN IV    EBCD  SUM 0020
C     TO 1.INITIALIZ E  2.SUMMARIZE  3.PRINT                            SUM 0030
C     AT END OF EACH MONTE CARLO PASS                                   SUM 0040
CKILLF                                                                  SUM 0050
      COMMON/CKILLF/LVSHP,MVSHP,NVSHP,VULSHP(5,12,1),NUN,NUNMX,         SUM 0060
     1 KSTATE(100),PKLAST(100),PPROD(100),CUMWT(100),VULOST(100),       SUM 0070
     2 VUKILL(100),VWKILL(100),VULFS,VULFA,VULFM                        SUM 0080
C***                                                                    SUM 0090
      COMMON/DEVICE/ N1,N2,N3,N4,N5,N6,N7,N8,N9,N10,N11,N12             SUM 0100
      COMMON/INOUU/ IPR(16),JPAR(16), PAR(16),LABEL                     SUM 0110
      COMMON/INOUT/NLINE,NPAGE,DUMA(35), NCODE(19)                      SUM 0120
     1,IDUMB(72), NFLAG, NFLAG2                                         SUM 0130
CNAVIG                                                                  SUM 0140
      COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                  SUM 0150
     1      NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),     SUM 0160
     2      NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),     SUM 0170
     3           TTIME,NUMX,                                            SUM 0180
     4      KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),  SUM 0190
     5      KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)   SUM 0200
CPLAT   VS   SYSTEMS   (SUBSYSTEMS)                                     SUM 0210
      COMMON/CPLAT/NBP,NBPMX,NBSS,NBSSMX,NBSPP(15),NBPWS(45),           SUM 0220
     1  NAMBP(2,15),KBPT(15),NABSS(2,45),KBST(45),KBPXS(45,15),         SUM 0230
     2  BPX1(12,15),                                                    SUM 0240
     3              NRP,NRPMX,NRSS,NRSSMX,NRSPP(15),NRPWS(45),          SUM 0250
     4  NAMRP(2,15),KRPT(15),NARSS(2,45),KRST(45),KRPXS(45,15),         SUM 0260
     5  RPX1(12,15)                                                     SUM 0270
CSUMM                                                                   SUM 0280
      COMMON/CSUMM/ SUMWT(100),SUMLST(100),SUMVUK(100),SUMVWK(100)      SUM 0290
CFORMATS                                                                SUM 0300
 2000  FORMAT(6X,'SEATIDE  -  NAVAL ENGAGEMENT MODEL  -',' SUMMARY AFTESUM 0310
     1R RANDOMIZED PASS NO.=', I3// 6X, A4,' SUMMARY BY UNITS'/         SUM 0320
     2 6X,'UNIT  NAME',6X,'TYPE GRUP STAT  PK   EHE-LBS',               SUM 0330
     3 4X,'VALU LOST  VALU KILLED', 5X,'AVG EHE',6X,'AVG LOST',         SUM 0340
     4 3X,'AVG KILLED'/ )                                               SUM 0350
 2002  FORMAT(3X,I6,'.',2X,2A4,1X, 3I5, F6.3, F8.0,5(1X,F12.3) )        SUM 0360
 2004  FORMAT(1H0,5X,'TOTAL 'A4,'VALUE LOST THIS PASS=',G13.6/          SUM 0370
     1          6X,'TOTAL AVERAGE VALUE LOST IN FIRST ',I3,' PASSES=',SUM 0380
     2          G13.6)                                                  SUM 0390
CDATA                                                                   SUM 0400
      DATA  BLU,RED/'BLU ','RED '/                                      SUM 0410
      DATA  ROTATE,RAD/ 30., .0174533/                                  SUM 0420
      NSYTMX  = NBSSMX + NRSSMX                                         SUM 0430
      NSYT  = NBSS + NRSS                                               SUM 0440
      IF (KPAS.GT.1)   GO TO 12                                         SUM 0450
```

```
C***  INITIALIZE
       DO 10 I=1,NUNMX
       SUMWT(I) = 0.
       SUMLST(I) = 0.
       SUMVUK(I) = 0.
 10    SUMVWK(I) = 0.
   12 CONTINUE
C***  CUM & TRANSER
 20    DO 22 I=1,NUN
       SUMWT(I) = SUMWT(I) + CUMWT(I)
       SUMLST(I) = SUMLST(I) + VULOST(I)
 22    SUMVUK(I) = SUMVUK(I) + VUKILL(I)
       DO 24 I=1,NSYT
 24    SUMVWK(I) = SUMVWK(I) + VWKILL(I)
C***  PRINT SUMMARIES
       DO 50 KSIDE=1,2
       GO TO (32,34), KSIDE
 32    NU  = KBU
       KA  = 1
       KB  = KBU
       BOR = BLU
       ASSIGN 42 TO KS1
       GO TO 40
 34    NU = KRU
       KA  = KBU + 1
       KB  = KBU + KRU
       BOR = RED
       ASSIGN 44 TO KS1
 40    CALL PAGE
       WRITE(N6,2000)  KPAS, BOR
       ATOTLS = 0.
       TOTLST = 0.
       JU  = 0
       DO 48 KU=KA,KB
       JU = JU + 1
       GO TO KS1, (42,44)
 42    NAMA = NAMBU(JU,1)
       NAMB = NAMBU(JU,2)
       KTY  = KBUK(2,JU) / 10000
       KGR  = KBUK(3,JU)
       KST  = KBUK(4,JU)
       GO TO 46
 44    NAMA = NAMRU(JU,1)
       NAMB = NAMRU(JU,2)
       KTY  = KRUK(2,JU) / 10000
       KGR  = KRUK(3,JU)
       KST  = KRUK(4,JU)
 46    PK  = PKLAST(KU)
       AEHF= SUMWT(KU) / KPAS
       AVL = SUMLST(KU) / KPAS
       AVUK= SUMVUK(KU) / KPAS
       ATOTLS= ATOTLS + AVL
       EHF = CUMWT(KU)
       VL  = VULOST(KU)
       VUK = VUKILL(KU)
```

```
      TOTLST= TOTLST + VL
      WRITE(N6,2002) JU,NAMA,NAMB,KTY,KGR,KST,PK,EHE,VL,VUK,
    1               AEHE,AVL,AVUK
 48   CONTINUE
      WRITE(N6,2004)  BOR,TOTLST,KPAS,ATOTLS
 50   CONTINUE
C*** ROTATE
   60 CEP=5.
      KTRAN=MOD(KPAS,3)+1
      KA=KPAS/3+1
      A=ROTATE
      AR=XNRN(IDUM)*A/10.
      XC=RA
      YC=RB
      XR=XNRN(IDUM)*CEP*.849
      YR=XNRN(IDUM)*CEP*.849
      IF(KA.GE.2) GO TO 62
      AR=0.
      XR=0.
      YR=0.
   62 A=A+AR
      IF(KTRAN.EQ.3) A=-2.*A
      JA=A+.01
      A=JA
      SINA=SIN(.0174533*A)
      COSA=COS(.0174533*A)
      DO 66 K=1,NRG
      DO 66 I=2,8
      X=RGC(I,2,K)
      Y=RGC(I,3,K)
      JHV=1000.*(RGC(I,5,K)+.0004)
      JH=JHV/1000
      JV=MOD(JHV,1000)
      XA=X-XC
      YA=Y-YC
      RGC(I,2,K)=XA*COSA+YA*SINA+XC+XR
      RGC(I,3,K)=-XA*SINA+YA*COSA+YC+YR
      JH=JH+JA
      IF(JH.LT.0) JH=JH+360
      IF(JH.GE.360) JH=JH-360
      RGC(I,5,K)=FLOAT(JH*1000+JV)/1000.
   66 CONTINUE
      RETURN
      END


       BLOCK  DATA
      REAL*8 III,ZZZ,XXX,PPP
      COMMON/URN1/ III,ZZZ,XXX,PPP
      COMMON/CGUID/ SDXP(2),SDYP(2),SKXP(2),SKYP(2),SEXP(2),SEYP(2),
     1SKX(2),SKY(2),SDRXT(2,3),SDRYT(2,3),SDRXDT(2,3),SDRYDT(2,3),THETA
      COMMON/BATTRY/ NBATT,MISRAD(2,10)
      COMMON /SONCRV/ IPA,IBB,IDA,IDB,IBOTTM,IACTIV,SRANGE(17),
    1       SMERIT(17)
```

```
      COMMON/ECONST/ NSHIP,NAIR,NSUB,NVSEA,NVALT,PDTMIN,PDMIN,           BLKA0090
     1     AIRCPT,FPNM,HMIN,PI,TWOPI,IENV,ISCAN,IPRAD,IMISC(35)          BLKA0100
       COMMON/DEVICE/ND(12)                                              BLKA0110
       COMMON/INOUU /IPR(16),JPAR(16),PAR(16),LABEL                      BLKA0120
      DATA III,ZZZ,XXX,PPP/4194305.D0,0.D0,.3243D0,2147483647D0/         BLKA0130
      DATA SDXP,SDYP,SKXP,SKYP,SEXP,SEYP,SKX,SKY,SDRXT,SDRYT,SDRXDT,     BLKA0140
     1SDRYDT/0.6E-4,0.3E-4,0.6E-4,0.3E-4,0.02,0.01,0.02,0.01,0.01,0.005,BLKA0150
     20.01,0.005,0.03,0.015,0.03,0.015,0.,600.,600.,6000.,600.,6000.,0.,BLKA0160
     3600.,600.,6000.,600.,6000.,0.,0.,0.25,1.,1.,3.,0.,0.,0.25,1.,1.,  BLKA0170
     43./,THETA/30./                                                     BLKA0180
      DATA NBATT/7/                                                      BLKA0190
      DATA MISPAD/6711,7713, 6712,7711, 6713,7712, 6714,7714,            BLKA0200
     1     8711,9711, 8713,9713, 8712,9712     /                         BLKA0210
      DATA IBA,IBB,IDA,IDB,IBOTTM,IACTIV/1,13,14,17,1,0/                 BLKA0220
      DATA SRANGE/2.,5.,7.7,20.,30.,40.,50.,60.,80.,100.,120.,           BLKA0230
     1     150.,180.,2.,5.,7.7,18./                                      BLKA0240
      DATA SMERIT/62.,80.,92.,94.,95.,97.,101.,107.,108.,110.,           BLKA0250
     1     115.,116.,120.,62.,80.,92.,120./                              BLKA0260
      DATA NSHIP,NAIR,NSUB,NVSEA,NVALT,IPRAD,ISCAN,IENV                  BLKA0270
     1     /1,2,3,6,9,0,3,4/                                             BLKA0280
      DATA PDTMIN,PDMIN,AIRCPT,FPNM,HMIN,PI,TWOPI                        BLKA0290
     1     /.5,.5,1.,6080.,.0033,3.141593,6.283186/                      BLKA0300
      DATA IMISC/35*0/                                                   BLKA0310
       DATA  ND/1,2,3,4,5,6,7,8,9,10,11,12/                              BLKA0320
       DATA  IPR/1,0,1,1,12*1/                                           BLKA0330
       DATA JPAR/16*1/                                                   BLKA0340
       DATA  PAR/16*1.0/                                                 BLKA0350
       DATA LABEL/ 1/                                                    BLKA0360
      END                                                                BLKA0370


       BLOCK DATA                                                        BLKB0010
C   PGM=NEM.  EVAN COTTEN. VER.3  7-26-73IBM 370   FORTRAN IV.    EBCD   BLKB0020
C      BLOCK  B                                                          BLKB0030
       COMMON/CNAVIG/ NGMX,BE,BF,RE,RF,                                  BLKB0040
     1     NBG,BA,BB,BC,BD,KBGN(18),KBGK(18),BGC( 8,6,18),NBU(18),       BLKB0050
     2     NRG,RA,RB,RC,RD,KRGN(18),KRGK(18),RGC( 8,6,18),NRU(18),       BLKB0060
     3         TTIME,NUMX,                                               BLKB0070
     4     KBU,KBUK(4,50),BREL(4,50),BXYZ(50,7),NAMBU(50,2),BV(50,8),    BLKB0080
     5     KRU,KRUK(4,50),RREL(4,50),RXYZ(50,7),NAMRU(50,2),RV(50,8)     BLKB0090
CRADAR                                                                   BLKB0100
       COMMON/CRADAR/NBSR,NBTR,NBRMX,NBJ,NBJMX,                          BLKB0110
     1   BRAD(36,17),BENV(6,2),BETC(4,2),BTAR(6,3),BJAM(6,3,2),          BLKB0120
     2               NRSR,NRTR,NRRMX,NRJ,NRJMX,                          BLKB0130
     3   RRAD(36,15),RENV(6,2),RETC(4,2),RTAR(6,3),RJAM(6,3,2)           BLKB0140
CSASYS                          15     4     4     2     4    20         BLKB0150
       COMMON/CSASYS/ NSASYS,NSASMX,NBGUN,NBSAM,NRGUN,NRSAM, LASA,       BLKB0160
     1    NAMSA(2,15),KSATYP(15),SASYS(20,15)                            BLKB0170
C                                                                        BLKB0180
       DATA  NGMX,NUMX/18,50/, NBG,NRG,KBU,KRU/4*0/                      BLKB0190
       DATA  BA,BB,BC,BD,RA,RB,RC,RD  / 8*.0/                            BLKB0200
       DATA  KBGN,KBGK,NBU,KRGN,KRGK,NRU/108*0/                          BLKB0210
       DATA  KBUK,KRUK/400*0/                                            BLKB0220
       DATA  BGC,RGC/1728*.0/, BREL,RREL/400*.0/, BXYZ,RXYZ/700*.0/      BLKB0230
      DATA   BV,RV/800*.0/                                                  BLKB0240
C                                                                           BLKB0250
      DATA   NBRMX,NRRMX,NBJMX,NRJMX/ 17,15, 2, 2/                          BLKB0260
      DATA   BENV/4.,0.,0.,1.,1.E-6,0.,4.,0.,0.,1.,1.E-8,0./                BLKB0270
      DATA   RENV/4.,0.,0.,1.,1.E-6,0.,4.,0.,0.,1.,1.E-8,0./                BLKB0280
      DATA   BETC/ 50.,      100.,0.,1.,150.,      150.,0.,1./              BLKB0290
      DATA   RETC/ 12160.,      100.,0.,1.,24320.,      150.,0.,1./         BLKB0300
      DATA   BTAR/ 500.,1000., 20., 20., 2*0.,                              BLKB0310
     1            675.,5000., 20., 20., 2*0.,                               BLKB0320
     2           1000.,10000.,20., 20., 2*0./                               BLKB0330
      DATA   RTAR/ 13.7,   1.,  608., 533.2, 2*0.,                          BLKB0340
     1            30.,    10.,24320., 533.2, 2*0.,                          BLKB0350
     2           500., 1000.,   20.,   20., 2*0./                           BLKB0360
      DATA   BJAM/ 36*0./                                                   BLKB0370
      DATA   RJAM/2000., 0.,25.E+6,1.  ,200.,225.,                          BLKB0380
     1  1000., 0.,70.E+6,1.  ,2965.,3035.,                                  BLKB0390
     2  1000., 0.,100.E+6,1.  ,3450.,3550.,                                 BLKB0400
     3   500., 0., 75.E+6,1.  ,5390.,5465.,                                 BLKB0410
     4   500., 0., 75.E+6,1.  ,5465.,5540.,                                 BLKB0420
     5   500., 0., 75.E+6,1.  ,5540.,5615. /                                BLKB0430
      DATA   SASYS/ 300*0./                                                 BLKB0440
      END                                                                   BLKB0450


C     SUBROUTINE RADAR(RAD,ENV,ETC,TAR,AMJ,TRAD,RANGE,PDA,ANG,DYN,ITST,     RADA0010
C    1IRNG,IPRNT)                                                           RADA0020
C  MODS.CALL SIGOS=3030.                              VER.2    11-25-74     RADA0030
C*********************************************************************     RADA0040
C                                                                    *     RADA0050
C                       ADRAD MOD 2 COMPUTER PROGRAM                 *     RADA0060
C                                                                    *     RADA0070
C*********************************************************************     RADA0080
C                                                                           RADA0090
C                                                                           RADA0100
C     DEFENSIVE RADAR PERFORMANCE AGAINST AIRBORNE TARGETS                  RADA0110
C     CASE 1 AIRBORNE TARGET WITHOUT ECM                                    RADA0120
C     CASE 2 AIRBORNE TARGET WITH COLOCATED NOISE JAMMER                    RADA0130
C     CASE 3 AIRBORNE TARGET WITH STANDOFF JAMMER                           RADA0140
C**************************************************************************RADA0150
C                                                                           RADA0160
C     INPUT PARAMETERS REQUIRED IN THE NAMELIST ARE AS FOLLOWS              RADA0170
C                                                                           RADA0180
C                       RADAR PARAMETERS                                    RADA0190
C                                                                           RADA0200
C     XLAMDA = RECEIVER CENTER FREQUENCY WAVELENGTH, CENTIMETERS            RADA0210
C     PWR = GENERATED RADAR POWER, WATTS                                    RADA0220
C     GTDB = RADAR TRANSMIT ANTENNA GAIN, DB                                RADA0230
C     GRDB = RECEIVER ANTENNA GAIN (MAIN LOBE), DB                          RADA0240
C     SNRDB=                                                                RADA0250
C     XNFDB = RADAR RECEIVER NOISE FIGURE, DB                               RADA0260
C     BR = RADAR RECEIVER BANDWIDTH, HERTZ                                  RADA0270
C     XLTDB = SYSTEM LOSSES AHEAD OF RF AMPLIFIER, DB                       RADA0280
C     XLRDB = RECEIVER LOSSES SUCH AS EFFICIENCY OR ANTENNA PATTERN, DB     RADA0290
C     XLSDB = RADAR SYSTEM LOSSES, DB                                       RADA0300
```

```
C     MOD= TYPE OF MODULATION USED                                      RADA0310
C           MOD = 1  UNCOMPRESSED PULSE                                 RADA0320
C           MOD = 2  PULSE COMPRESSION                                  RADA0330
C           MOD = 3  PULSE BURST                                        RADA0340
C           MOD = 4  CW                                                 RADA0350
C           MOD = 5  FM-CW                                              RADA0360
C           MOD = 6  MTI                                                RADA0370
C           MOD = 7  PULSE DOPPLER                                      RADA0380
C     TAU = RADAR PULSE WIDTH, MICRO SECONDS                            RADA0390
C     PRF = PULSE REPETITION RATE, PULSES PER SECOND                    RADA0400
C     FAN = THE NUMBER OF PULSES WHICH CHANGE IN FREQUENCY BY           RADA0410
C           THE RECIPROCAL OF THE PULSE WIDTH                           RADA0420
C     PCRAT = PULSE COMPRESSION RATIO, NUMERIC                          RADA0430
C     ITYPE= TYPE OF NORMALIZED ANTENNA GAIN FUNCTION.                  RADA0440
C           = 1 IS SYMMETRICAL SINX/X PATTERN                           RADA0450
C           = 2 IS A CSC SQUARED FUNCTION                               RADA0460
C     PHIZER = MINMUM ANGLE OFF ANTENNA BORESIGHT AXIS AT               RADA0470
C              WHICH THE PATTERN FUNCTION BECOMES A CSC SQUARED         RADA0480
C              FUNCTION                                                 RADA0490
C     RATEL = ANTENNA AZIMUTH SCAN RATE, DEGREES PER SECOND             RADA0500
C     BWA = ANTENNA AZIMUTH BEAMWIDTH, DEGREES                          RADA0510
C     BWE = BEAM WIDTH OF ANTENNA IN ELEVATION, DEGREES                 RADA0520
C     TILT = MAIN BEAM ELEVATION ANGLE (DEG)                            RADA0530
C     TA = ANTENNA TEMPERATURE, DEGREES K                               RADA0540
C     EINSTR = INSTRUMENT ERROR IN ANGLE IN MRAD                        RADA0550
C     TI = INTEGRATION OR SMOOTHING TIME DURING TRACK                   RADA0560
C     AZ = AZIMUTH COMPONENT OF THE SFOV , DEGREES                      RADA0570
C     EL = ELEVATION COMPONENT OF SFOV (DEGREES)                        RADA0580
C     TF = FRAME TIME,SECONDS (TIME TO SEARCH THE SFOV)                 RADA0590
C                                                                       RADA0600
C                      ENVIRONMENT AND DETECTION CRITERIA               RADA0610
C                                                                       RADA0620
C     ISS = SEA STATE NUMBER                                            RADA0630
C     RATE = RAINFALL RATE IN MM/HR                                     RADA0640
C     ALPHA = RAINFALL ATTENUATION IN DB/METER AT SPECIFIED RAINFALL    RADA0650
C             RATE AND WAVELENGTH (XLAMDA)                              RADA0660
C     NSW = SWERLING FACTOR                                             RADA0670
C     PFA = PROBABILITY OF FALSE ALARM                                  RADA0680
C                                                                       RADA0690
C                      SEQUENCE DEPENDENT                               RADA0700
C                                                                       RADA0710
C     HI = RADAR HEIGHT (FT)                                            RADA0720
C     SOR = JAMMER STAND-OFF RANGE IN NAUTICAL MILES                    RADA0730
C     JAM = 0 FOR NO JAMMING                                            RADA0740
C         = 1 FOR JAMMER COLOCATED WITH TARGET                          RADA0750
C         = 2 FOR STAND-OFF JAMMER                                      RADA0760
C     NMODE = 1 FOR RADAR OPERATING IN SURVEILLANCE(SEARCH) MODE        RADA0770
C     NMODE = 2 FOR RADAR OPERATING IN TRACK MODE (TILTR=ELVA)          RADA0780
C             AND USING SIMULTANEOUS LOBING(MONOPULSE)                  RADA0790
C     NMODE = 3 FOR RADAR OPERATING IN TRACK MODE(TILTR = ELVA)         RADA0800
C             AND USING CONICAL SCAN                                    RADA0810
C     NMODE = 4 FOR RADAR OPERATING IN LIMITED VOLUME SEARCH            RADA0820
C                                                                       RADA0830
C                          TARGET CHARACTERISTICS                       RADA0840
C                                                                       RADA0850
```

```
C       ALTGT = PROJECTED LENGTH OF TARGET ORTHOGONAL TO THE RADAR
C               LINE-OF-SIGHT, FEET
C       SIGMAT = RADAR CROSS SECTION OF THE TARGET, SQUARE METERS
C       H2 = TARGET HEIGHT (FT)
C       VRELK = RELATIVE CLOSING VELOCITY, KNOTS
C       DR = RANGE IN NMI AT WHICH THE TARGET BEGINS ITS DESCENT
C       NOTE  IF TARGET MAINTAINS CONSTANT ALTITUDE, SET DR = 0.0
C       AK20 = DESCENT CONSTANT  MAINTAINS CONSTANT H/R DURING
C             FLIGHT
C
C                             JAMMER CHARACTERISTICS
C
C       PJ = GENERATED JAMMER POWER, WATTS
C       GJDB = TRANSMIT ANTENNA GAIN OF JAMMERSIGNAL, DB
C       BJ = JAMMER SIGNAL BANDWIDTH, ALWAYS EQUAL TO OR GREATER THAN BR
C       XLJDB = LOSSES IN THE JAMMER TRANSMITTER CHAIN SUCH AS ANTENNA
C               AND WAVEGUIDE LOSSES, DB
C
C
        SUBROUTINE RADAR(RAD,ENV,ETC,TAR,AMJ,TRAD,RANGE,PDA,ANG,DYN,ITST,
       1IRNG,IPRNT)
         COMMON/INPUT/NLINE,NPAGE,PCODE(20),IDUM(108)
        DIMENSION DAT(49),RAD(31),ENV(6),ETC(4),TAR(6),AMJ(4),TRAD(10),
       1RANGE(10),PDA(11),ANG(10),DYN(10)
        COMMON /BLK0/BETA,ALFA,XLAMDA,PWR,GTDB,GRDB,SNRODB,XNFDB,BR,XLTDB,
       1XLRDB,XLSDB,AMOD,TAU,PRF,FAN,PCRAT,PHIMAX,TYPE,PHIZER,RATE1,BWA,
       2BWE,TILT,TA,EINSTR,TI,AZ,EL,TF,XK,AISS,RATE,ALPHA,ANSW,PFA,HR,HI,
       3SOR,AJAM,ANMODE,ALTGT,SIGMAT,H2,VRELK,DR,AK20,PJ,GJDB,BJ,XLJDB,
       4ISS,ITYPE,NMODE,VREL
        COMMON /BLK3/ FF, RC, DX, IHACK ,ELVA, CC, REC(10), TC
        EQUIVALENCE (DAT(1),XLAMDA)
 1001 FORMAT(6X,19HI N P U T   D A T A,25X31HR A D A R   P A R A M E T E
     1 R S/6X7HXLAMDA=1PF10.3,11H,    PR     =1PF10.3,10H,   GTDB =,1PE10.
     23,10H,    GRDB =,1PF10.3,11X,           1PF10.3,10H,   XNF  =,1PE10.
     33,1H,/6X7HBR     =1PF10.3,11H,    XL     =1PE10.3,10H,   XLR  =1PF10.
     43,10H,    XLSDB=,1PF10.3,11H,   AMOD   =1PF10.3,10H,   TAU  =,1PF10.
     53,1H,/6X7HFR     =1PF10.3,11H,    FAN    =1PE10.3,10H,   PCRAT=1PF10.
     63,10H,    TYPE =,1PF10.3,11H,   PHIZER=1PF10.3,10H,   RATE1=,1PF10.
     73,1H,/6X7HBWE    =1PF10.3,11H,    BWD    =1PF10.3,10H,   TILT =1PE10.
     83,10H,    TA   =,1PE10.3,11H,   EINSTR=1PE10.3,10H,   TI   =,1PE10.
     93,1H,/6X7HAZ     =1PF10.3,11H,    EL     =1PE10.3,10H,   TF   =1PF10.
     13,1H,/30X67HE N V I R O N M E N T   A N D   D E T E C T I O N   C
     2R I T E R I A/6X7HAISS  =1PF10.3,11H,   RATE  =1PE10.3,10H,   ALPH
     3A=,1PF1C.3,10H,   ANSW =,1PF10.3,11H,   PFA   =1PE10.3,1H,/48X35HS
     4 E Q U E N C E   D E P E N D E N T/6X7HHI     =1PF10.3,11H,   SJDB
     5 =1PF10.3,10H,   SOR  =,1PF10.3,10H,   AJAM =,1PE10.3,11H,   ANMOD
     6E=1PF10.3,1H,/44X43HT A R G E T   C H A R A C T E R I S T I C S/6X
     77HALNGTH=1PF10.3,11H,   SIGMAT=1PF10.3,10H,   H2   =,1PF10.3,10H,
     8  VREL =,1PF10.3,11H,   DR     =1PF10.3,10H,   AK20 =,1PF10.3,1H,/4
     94X43HJ A M M E R   C H A R A C T E R I S T I C S)
 1002 FORMAT(6X7HPJ     =1PE10.3,11H,   GJDB  =1PF10.3,10H,   BJ   =,1PF1
     10.3,10H,   XLJ  =,1PF1C.3,1H,/)
 1003 FORMAT(12X4HTIME11X5HRANGE8X9HPROB.DET.7X7HHANDOFF8X7HDYNAMIC)
 1004 FORMAT(12X4HTIME11X5HRANGE8X9HPROB.DET.6X9HMAX TRACK7X7HDYNAMIC)
 1005 FORMAT(12X4HSEC.12X3HNMI9X9HPER   LOOK5X11HERROR(MRAD)5X9HRANGE(DB)
     1/)
 1006 FORMAT(4X1P5E15.3)
      XNSCAN=0.
      RCMIN=.5*(1.+ABS(ETC(1)-TAR(3))/6080.)
      RCSUM=0.
      RCA(IRNG+1)=0.
      ITST=0
C--- QUICK OUT FOR RADAR HORIZON LIMITATION -----------
      IF(1.2289*(SQRT(ETC(1))+SQRT(TAR(3))) .GE. RANGE(1)) GO TO 18
      ITST=-1
      RETURN
   18 DO 180 I = 1, IRNG
      RCA(I) = 0.
  180 RFC(I) = RANGE(I)
      TRAD(1)=TRAD(1)*3600.
      DO 1 I = 1, 29
      K = I + 2
    1 DAT(I) = RAD(K)
      DO 4 I = 1, 6
      DAT(I+39) = TAR(I)
      DAT(I+29) = ENV(I)
      GO TO (2,2,2,2,4,4), I
    2 DAT(I+45) = AMJ(I)
      DAT(I+35) = ETC(I)
    4 CONTINUE
      MOD = AMOD + .1
      ISS = AISS + .1
      NSW = ANSW + .1
      JAM = AJAM + .1
      ITYPE = TYPE + .1
      NMODE = ANMODE + .1
      IF (IPRNT - 1) 300, 300, 200
  200 IF (NLINE - 38) 220, 220, 210
  210 CALL PAGE
  220 IF (IPRNT - 1) 300, 240, 230
  230 WRITE(6,1001) XLAMDA,PWR,GTDB,GRDB,SNRODB,XNFDB,BR,XLTDB,XLRDB,
     1XLSDB,AMOD,TAU,PRF,FAN,PCRAT,TYPE,PHIZER,RATE1,BWA,BWE,TILT,TA,
     2EINSTR,TI,AZ,EL,TF,AISS,RATE,ALPHA,ANSW,PFA,H1,SJDB,SCR,AJAM,
     3ANMODE,ALTGT,SIGMAT,H2,VRELK,DR,AK20
      WRITE(6,1002)  PJ,GJDB,BJ,XLJDB
      NLINE = NLINE + 15
  240 GO TO (250, 260, 260, 250), NMODE
  250 WRITE(6,1003)
      GO TO 270
  260 WRITE(6,1004)
  270 WRITE(6,1005)
      NLINE = NLINE + 3
  300 CC = 1.
      DEGRAD = 57.29578
      C = 9.843E8
      AK = 1.38E-23
      RE2= 1.69893E7
      XL  = 10. ** (.1*XLTDB)
      XNF = 10. ** (.1*XNFDB)
      ALNGTH = ALTGT * .304801
      VREL = VRELK * 1.6878                                             RADA1960
      IF(VREL .LT. .1) VREL=.1                                          RADA1970
      H2TEMP=H2                                                         RADA1980
      PRTEMP = PWR                                                      RADA1990
      TAUTMP = TAU                                                      RADA2000
      PNODET = 1.                                                       RADA2010
      GO TO (6, 5, 5, 7), NMODE                                         RADA2020
    5 TOT = TI                                                          RADA2030
      GO TO 7                                                           RADA2040
    6 TOT = BWA / RATE1                                                 RADA2050
      TC = 360. / RATE1                                                 RADA2060
      GO TO 8                                                           RADA2070
    7 TC = 6.                                                           RADA2080
    8 PHORAD = PHIZER/57.2957795D0                                      RADA2090
C     COMPUTE SYSTEM NOISE TEMPERATURE                                  RADA2100
      TS = TA/XL +((XL-1.)/XL)*290. + (XNF-1.)*290.                     RADA2110
C     COMPUTE SIGNAL PROCESSING GAIN, SPGDB                             RADA2120
C     SPGDB SIGNAL PROCESSING GAIN BASED ON SNRODB = 22.4 FOR ONE PULSE RADA2130
C     PD = 0.9 AND PFA = 1.E-8  SWERLING CASE 1 AND SQ LAW DETECTOR     RADA2140
      IF (MOD .NE. 2) GO TO 11                                          RADA2150
      PWR = PWR * PCRAT                                                 RADA2160
      TAU = TAU / PCRAT                                                 RADA2170
   11 GAIN = 10.**(GTDB/10.)                                            RADA2180
      PRDB = 10. * ALOG10(PWR)                                          RADA2190
C CONVERT XLAMDA IN CM TO METERS                                       RADA2200
      D1 = .01*XLAMDA                                                   RADA2210
      WSQRDB = 20.*ALOG10(D1)                                           RADA2220
      XNUMDB = PRDB + GTDB + GRDB + WSQRDB                              RADA2230
      P3DB = 33.                                                        RADA2240
      DENDB = P3DB + XLSDB                                              RADA2250
      XKDB = XNUMDB - DENDB                                             RADA2260
      RCSDB = 10.*ALOG10(SIGMAT)                                        RADA2270
      THET1 = BWA / 57.29578                                            RADA2280
      THET2 = BWE / 57.29578                                            RADA2290
      XK2 = THET1*THET2                                                 RADA2300
      XK1 = 150.*TAU                                                    RADA2310
      XRN = ((RATE**1.6)*5.6919*1.6E-6)/XLAMDA**4                       RADA2320
      XKLG = XKDB/10.                                                   RADA2330
      XK = 10.**XKLG                                                    RADA2340
      H1MET = H1*.3048                                                  RADA2350
      RH1 = 1.2289 * SQRT(H1)                                           RADA2360
      HALFAZ = THET1/2.                                                 RADA2370
      HALFEL = THET2/2.                                                 RADA2380
      T1 = TAN(HALFAZ)                                                  RADA2390
      T2 = TAN(HALFEL)                                                  RADA2400
      TILTR = TILT/57.29578                                             RADA2410
      TILTE = TILTR                                                     RADA2420
      PJDB = 10.*ALOG10(PJ)                                             RADA2430
      BRDB = 10.*ALOG10(BR)                                             RADA2440
      BJDB = 10.*ALOG10(BJ)                                             RADA2450
      GO TO (13, 13, 13, 12), NMODE                                     RADA2460
   12 OMEGA = 4.*3.1416/GAIN                                            RADA2470
      ELANG = EL/DEGRAD                                                 RADA2480
      ANGMAX = TILTE + ELANG/2.                                         RADA2490
      ANGMIN = TILTE - ELANG/2.                                         RADA2500
      AZANG = AZ/DEGRAD
      SFOV = 3.1416*TAN(ELANG/2.)*TAN(AZANG/2.)
      TDS = TF*OMEGA/SFOV
      TC1 = TDS
   13 TC2 = (C*TAU*1.E-6)/(4.*VREL)
      TC = AMIN1(TC1, TC2)
      XN = PRF * TC
      N = INT(XN)
      FA = -ALOG10(PFA)
      SPGDB = 9.2*ALOG10(XN) -2.6*ALOG10(XN)
C   COMPUTE DETECTION RANGE IN THE ABSENCE OF ECM
      IHACK = 0
      CALL FFACT
      IHACK = 1
      DO 20  I = 1, IRNG
      IF (I .NE. 1) TRAD(I) = TRAD(I-1) + TC
      CALL FFACT
      IF(IHACK.EQ.3) GO TO 50
      FFDB = 10.*ALOG10(FF)
      RMET = RC*1.852E3
      IF (NMODE .NE. 4) GO TO 14
      IF (TILT) 16, 15, 15
   16 IF (ELVA .GT. ANGMAX) GO TO 135
      IF (ELVA .LT. ANGMIN) GO TO 50
      GO TO 14
   15 IF(ELVA.GT.ANGMAX) GO TO 50
      IF(ELVA.LT.ANGMIN) GO TO 135
   14 EGLINT = (350.*ALNGTH)/RMET
      IF(EGLINT.LE..001) EGLINT = 0.
      EF = EGLINT**2 + EINSTR**2
      IF (RC .LE.DR) H2 = AK20*RMET*3.28084
      RMETDB = 10.*ALOG10(RMET)
      PSTDB = XKDB + RCSDB + 4.*FFDB -4.*RMETDB
C     COMPUTE RAIN CLUTTER RETURN, PRCA (WATTS)
      VRN = .785*(RMET**2)*XK2*XK1
      PRC =  (XK*VRN*XRN*(FF**4))/((RMET**4)*FAN)
      PN = AK*TS*BR
      ATTDB = 2.*ALPHA*RMET
      AT1 = -0.1*ATTDB
      ATT = 10.**AT1
      PSTADB = PSTDB - ATTDB
      PRCA = PRC*ATT
      PSTALG = PSTADB*0.1
      PSTA = 10.**PSTALG
C   COMPUTE SEA CLUTTER RETURN, PSCA (WATTS)
      F4 = H1MET/RMET
      F5 = RMET/RE2
      IF (RC .GT.RH1)  GO TO 71
      ARG = F4-F5
      IF(ARG.GE.1.) GO TO 50
      PSI = ARSIN(ARG)
      XK4 = TAN(PSI)
   45 ACL = 2.*RMET*XK1*T1/COS(PSI)
      CALL SIGOS(ISS, XLAMDA, PSI, SIGODB)
C   DELETE CARDS DOWN TO STATEMENT  70
```

```
   70 CONTINUE
      ACLDB = 10.*ALOG10(ACL)
      SIGCDB = ACLDB + SIGODB
   71 ARG1 = F4 + F5
      PHI = ARSIN(ARG1)
      IF(NMODE.GT.1) TILTR = ELVA
      BWDR = BWF / 57.29578
      IF (RC .GT.RH1)  GO TO 72
      IF ( ITYPE .EQ. 1) CALL FGAIN(GD,TILTR,-PHI,BWDR)
      IF ( ITYPE .EQ. 2) CALL FGCSC(GD,TILTR,-PHI,PHORAD)
C     COMPUTE ANTENNA PATTERN FACTOR,FC, TO BE USED IN THE
C     CALCULATION OF SEA CLUTTER RETURN.
C     NOTE   NO PROPAGATION FACTOR MUST BE COMPUTED, SINCE IT IS
C     INCLUDED IN THE MEASURED VALUES OF SIGMO.
      FC = GD
      F1 = FC**4
      FCDB4 = 10.*ALOG10(F1)
      PSCDB = XNUMDB + SIGCDB - DENDB - 40.*ALOG10(RMET) + FCDB4
      ATT1DB = 2.*ALPHA*RMET
      PSCADB = PSCDB - ATT1DB
      PSCALG = 0.1*PSCADB
      PSCA =(10.**PSCALG)/FAN
      GO TO 73
   72 PSCA = 0.0
   73 JM = JAM + 1
      GO TO (75, 80, 85), JM
   75 AJR = 0.
      GO TO 102
   80 RJ = RMET
      GO TO 90
   85 RJ = SCP * 1852.
   90 ATT2DB = ALPHA*RJ
      AJUPDB = PJDB + GJDB + BPDB + GRDB + WSQRDB -ATT2DB + 2.*FFDB
      AJDNDB = 20.*ALOG10(12.56) + 20.*ALOG10(RJ) + BJDB + XLJDB
     1 +XLRDB
      AJRDB = AJUPDB - AJDNDB
      AJRDLG = AJRDB*.1
      AJR = 10.**AJRDLG
  102 AJLO = PN + PRCA + AJR + PSCA
      XLOWDB = 10.*ALOG10(AJLO)
      SNR1DB = PSTADB-XLOWDB
      IF(SNR1DB.LE.-26.)      SNR1DB=-26.
      SNRDB = SNR1DB + SPGDB
      SNR = 10.**(0.1*SNRDB)
      CALL MAPSWR(SNR1DB,N,FA,NSW,PD)
      IF(NMODE.EQ.4) PDJ = PD
C     IF(NMODE.EQ.4) PNODET = (1. - PDJ)*PNODET
C     IF(NMODE.EQ.1) PNODET = (1. - PD)*PNODET
      PDCUM = 1. - PNODET
      STOTAL = PSTA + PSCA + PRCA + AJR
      DYN1 = STOTAL/PN
      DYNRDB = 10.*ALOG10(DYN1)
  107 GO TO (120, 121, 122, 121), NMODE
  120 THT = THET1
      FC = 0.5
      FP = 0.1                                                          RADA3610
      GO TO 124                                                         RADA3620
  121 EC = 1.                                                           RADA3630
      GO TO 123                                                         RADA3640
  122 EC = 2.                                                           RADA3650
  123 THT = THET2                                                       RADA3660
      FP = 1.5                                                          RADA3670
  124 CALL ANGERR(VAR,SNR,THT,EC,FP)                                    RADA3680
      PDSUM=PDSUM+PD                                                    RADA3690
      EC = SQRT(VAR + EE)                                               RADA3700
      RANGE(I) = RC                                                     RADA3710
      PDA(I) = PD                                                       RADA3720
      ANG(I) = EC                                                       RADA3730
      DYN(I) = DYNRDB                                                   RADA3740
      IF(IPRNT.EQ.0) GO TO 135                                          RADA3750
      WRITE(6,1006) TRAD(I),RC,PD,EC,DYNRDB                             RADA3760
      NLINE = NLINE + 1                                                 RADA3770
  135 RC = RC - DX                                                      RADA3780
      XNSCAN=XNSCAN+1.                                                  RADA3790
      IF(RC.LT.RCMIN) GO TO 50                                          RADA3800
   20 CONTINUE                                                          RADA3810
   50 CONTINUE                                                          RADA3820
      DO 400 I = 1, IRNG                                                RADA3830
  400 TRAD(I) = TRAD(I) / 3600.                                         RADA3840
      IF(XNSCAN.GT.0.) PDA(IRNG+1)=PDSUM/XNSCAN                         RADA3850
      H2=H2TEMP                                                         RADA3860
      PWR = PRTEMP                                                      RADA3870
      TAU = TAUTMP                                                      RADA3880
      RETURN                                                            RADA3890
      END                                                               RADA3900


      SUBROUTINE FFACT                                                  FFCT0010
C                                                                       FFCT0020
C MODS.1140,1580,1810.                                 VER.2   11-25-74 FFCT0030
C THIS SUBROUTINE CALCULATES PATTERN PROPOGATION FACTORS USING ROUTINES FFCT0040
C FROM NRL REPORT 7098.                                                 FFCT0050
C                                                                       FFCT0060
C MODIFICATIONS MADE BY G H COLBY  15 AUG 1972                          FFCT0070
C                                                                       FFCT0080
C                                                                       FFCT0090
C***********************************************************************FFCT0100
C                                                                       FFCT0110
C         REQUIRED INPUTS VIA NAMELIST INPUT ARE                        FFCT0120
C                                                                       FFCT0130
C                   H1         = RADAR HEIGHT (FT)                      FFCT0140
C                   H2         = TARGET HEIGHT (FT)                     FFCT0150
C                   XLAMDA     = WAVE LENGTH (CM)                       FFCT0160
C                   ISS        = SEA STATE NUMBER                       FFCT0170
C                   BWE        = BEAM WIDTH (DEG)                       FFCT0180
C                   TILT       = MAIN BEAM ELEVATION ANGLE (DEG)        FFCT0190
C       PHIZER = MINMUM ANGLE OFF ANTENNA BORESIGHT AXIS AT             FFCT0200
C                WHICH THE PATTERN FUNCTION BECOMES A CSC SQUARED       FFCT0210
C                FUNCTION                                               FFCT0220
```

```
C      ITYPE= TYPE OF NORMALIZED ANTENNA GAIN FUNCTION.
C      ITYPE = 1 IS SYMMETRICAL SINX/X PATTERN
C      ITYPE = 2 IS A CSC SQUARED FUNCTION
C     NMODE = 1 FOR RADAR OPERATING IN SURVEILLANCE(SEARCH) MODE
C      NMODE = 2 FOR RADAR OPERATING IN TRACK MODE (TILTR=ELVA)
C
C***********************************************************************
C
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*4 BETA,ALFA,XLAMDA,PWR,GTDB,GRDB,SNRODB,XNFDB,BR,XLTDB,XLRDB,
     1XLSDB,AMOD,TAU,PRF,FAN,PCRAT,PHIMAX,TYPE,PHIZER,RATE1,BWA,BWE,
     2TILT,TA,EINSTR,TI,AZ,EL,TF,XK,AISS,RATE,ELPHA,ANSW,PFA,HR,H1,SOR,
     3AJAM,ANMODE,ALTGT,SIGMAT,H2,VRELK,DR,AK20,PJ,GJDB,BJ,XLJDB,RH,FF,
     4DX,RC,ELVA,CE,RANGE,TC,FDIR,TILTR,BWDR,PHORAD,FREF,DEPA,SIN,SIM,
     5S1,S,VREL,T,TT
      COMMON /BLK0/BETA,ALFA,XLAMDA,PWR,GTDB,GRDB,SNRODB,XNFDB,BR,XLTDB,
     1XLRDB,XLSDB,AMOD,TAU,PRF,FAN,PCRAT,PHIMAX,TYPE,PHIZER,RATE1,BWA,
     2BWE,TILT,TA,EINSTR,TI,AZ,EL,TF,XK,AISS,RATE,ELPHA,ANSW,PFA,HR,H1,
     3SOR,AJAM,ANMODE,ALTGT,SIGMAT,H2,VRELK,DR,AK20,PJ,GJDB,BJ,XLJDB,
     4ISS,ITYPE,NMODE,VREL
      COMMON /BLK1/  XX(4), YY(4), AC, BC, CC, DC
      COMMON /BLK2/  T, TT
      COMMON /BLK3/ FF, RC, DX, IHACK, ELVA, CE, RANGE, TC
      DIMENSION     WHET(7), RANGE(10)
      DATA  WHET  /0.0,0.5,2.0,4.0,6.5,10.,16./
C
      IF (IHACK .GT. 0)        GO TO 100
      INDX     =  0
      PI       =  3.1415927D0
      RDN      =  57.2957795D0
      TUPI     =  6.2818531D0
      PI2      =  1.5707963D0
      FPNM     =  6076.1155D0
      FTPM     =  3.280833D0
      RE       =  4.D0 * 3440.D0 / 3.D0
      C        =  29977.6D0
      AJ       =  0.0D0
      E1       =  0.0D0
C
      TILTR    =  TILT /  RDN
      BWDR = BWE / RDN
      PHORAD = PHIZER/RDN
C
C SET WAVE HEIGHT
C
      IT       =  ISS     +  1
      WH       =  WHET(IT)  *  .35355339D0
C
C SET RADAR FREQUENCY
C
      FMHZ     =  C  /  XLAMDA
C
C SET T
C
      T        =  SQRT(H1 / H2)
      IF (T .GT. 1.0D0)         T  =  1.0D0 / T                         FECT0780
      TT        =  T * T                                                FECT0790
C                                                                       FECT0800
C TOTAL HORIZON DISTANCE                                                 FECT0810
C                                                                       FECT0820
      RH        = 1.2289 * (SQRT(H1) + SQRT(H2))                        FECT0830
      DX = TC * VREL / 6076.116                                         FECT0840
C                                                                       FECT0850
C QUARTER WAVELENGTH                                                    FECT0860
    2 W         = (XLAMDA / 100.D0) * FTPM                              FECT0870
      W4        =  W * .25D0                                            FECT0880
C                                                                       FECT0890
C FACTORS                                                               FECT0900
C                                                                       FECT0910
      FAC       = (2.0D0 * H1 * H2) / FPNM                              FECT0920
      TEMP      = (H2 - H1)  /  FPNM                                    FECT0930
      HH        =  TEMP    *  TEMP                                      FECT0940
      R         = (FAC / W4) *  1.2D0                                   FECT0950
      IF (R .GT. RH)     R  =  RH  *  .999D0                            FECT0960
   10 IF  ((AJ .GE. 0.999D0) .AND. (D1.GE. 0.999D0))  GO TO 20          FECT0970
C                                                                       FECT0980
C USE ROUND EARTH HACK                                                  FECT0990
C                                                                       FECT1000
      S         =  R  /  RH                                             FECT1010
      SIN       =  S                                                    FECT1020
      SIM       =  S  *  1.2D0                                          FECT1030
      CALL  INVERT(SIN,SIM,S1,S)                                        FECT1040
      S1 = AMIN1(S1,.999)                                               FECT1050
      SS1       =  S1 * S1                                              FECT1060
      SQ1       =((1.D0 - SS1)**2) + (4.D0 * SS1 * TT)                  FECT1070
      S2        = (DSQRT(SQ1)  -  1.D0  +  SS1) / (2.D0 * S1 * T)       FECT1080
      SS2       =  S2 * S2                                              FECT1090
      TEMP1     =  4.D0 * SS1 * S2 * T                                  FECT1100
      TEMP2     =  S  * (1.D0 - SS1) * (1.D0 + T)                       FECT1110
      SQ2       =  1.D0  +  (TEMP1 / TEMP2)                             FECT1120
      D1        =  1.D0  /  DSQRT(SQ2)                                  FECT1130
      AJ        = (1.D0  -  SS1 ) * (1.D0  -  SS2)                      FECT1140
      AK = ((1.D0-SS1) + TT*(1.D0-SS2))/(1.D0 + TT)                     FECT1150
      GO TO 30                                                          FECT1160
C FOR FLAT EARTH                                                        FECT1170
   20 AJ        =  1.D0                                                 FECT1180
      D1        =  1.D0                                                 FECT1190
      AK        =  1.D0                                                 FECT1200
C                                                                       FECT1210
   30 DELTA     = (FAC    /  R)      *  AJ                              FECT1220
      RSLANT    =  DSQRT((R*R)  +  HH)                                  FECT1230
      IF (DELTA .LT. W4)       GO TO 130                                FECT1240
   70 INDX      =  INDX  +  1                                           FECT1250
      IF (INDX .EQ. 2)           RTEST  =  RSLANT                       FECT1260
   90 PSI       =  DATAN(((H1+H2) / (FPNM * R)) * AK)                   FECT1270
      CALL  SEAREF(FMHZ,PSI,RHO,PHI)                                    FECT1280
      RATIO     = (DELTA / W)  +  (PHI / TUPI)                          FECT1290
      ITEMP     =  IDINT(RATIO)                                         FECT1300
      WHOLES    =  DFLOAT(ITEMP)                                        FECT1310
      RATIO1    =  RATIO - WHOLES                                       FECT1320
      ALPHA    =  TUPI   * RATIO1
      TEMP1    = (PI   *   WH   *   DSIN(PSI))/   W
      TEMP1    =   TEMP1 * TEMP1
      RUF = 0.D0
      AX = 8.D0*TEMP1
      IF(AX.LT.58.)RUF = DEXP(-AX)
C
C CALCULATE ELEVATION ANGLE TO THE TARGET
C
      TEMP1    =   H2  -  H1
      TEMP2    =   .6624D0   *   R *R
      ELVA     =   DATAN2 ((TEMP1-TEMP2),(R*FPNM))
      IF(NMODE.GT.1) TILTR = ELVA
C GAIN FACTOR DIRECT PATH
   55 IF (ITYPE .EQ. 1) CALL FGAIN(FDIR,ELVA,TILTR,BWDR)
      IF (ITYPE .EQ. 2) CALL FGCSC(FDIR,ELVA,TILTR,PHORAD)
C
C CALCULATE DEPRESSION ANGLE TO REFLECTION POINT
C
      H        =   H1     / FPNM
      TEMP1    =   DARSIN (RE * DCOS(PSI) / (RE + H))
      DEPA     =   PI2  -  TEMP1
C GAIN FACTOR REFLECTED PATH
      IF (ITYPE .EQ. 1) CALL FGAIN(FREF,DEPA,-TILTR,BWDR)
      IF (ITYPE .EQ. 2) CALL FGCSC(FREF,DEPA,-TILTR,PHORAD)
      D        =   D1   *   RHO   *   RUF   *   FREF   /   FDIR
C
      FF       =   DSQRT (1.D0 +(D*D) + (2.D0 * D* DCOS(ALPHA)))
      FF       =     ABS(FDIR)   *   FF
      IF (INDX .GT. 2)          GO TO 200
      ISUB     =   INDX   +  2
      XX( ISUB )     =  RSLANT
      YY(ISUB)       =   FF
      IF (INDX .EQ. 1)          GO TO 130
      RNM      =   2.D0    *   RH
      CALL DFRACT (H1, H2, RNM, FMHZ, FDB)
      FF       = (10.D0) ** (FDB * .05D0)
      XX(2)    =   RNM
      YY(2)    =.  FF
      RNM      =   1.1D0   *   RNM
      CALL DFRACT (H1, H2, RNM, FMHZ, FDB)
      FF       = (10.D0) ** (FDB * .05D0)
      XX(1)    =   RNM
      YY(1)    =   FF
      CALL CURVIT
      RC = RANGE(1)
      RETURN
  100 IF (RC   .LT.  RTEST)     GO TO 120
      FF       =   DC + RC * ( CC + RC * ( BC + RC * AC ))
C
      IF (FF .LE.1.E-6) FF = 1.E-6
      RETURN
C
C SET VALUE OF R FOR INTERFERENCE ZONE HACK
C
```

```
  120 TEMP1    = (RC *RC)  -   HH                                      EFCT1880
      IFACK    =  2                                                    EFCT1890
      IF (TEMP1 .LE. 0.D0)     GO TO 300                               EFCT1900
      P        =  DSQRT(TEMP1)                                         EFCT1910
      GO TO 10                                                         EFCT1920
C                                                                      EFCT1930
C                                                                      EFCT1940
  130 R        =  R  *   .99D0                                         EFCT1950
      GO TO 10                                                         EFCT1960
C                                                                      EFCT1970
  200 RETURN                                                           EFCT1980
  300 IFACK    =  3                                                    EFCT1990
      RETURN                                                           EFCT2000
      END                                                              EFCT2010


      SUBROUTINE MARSWR (SNDB,N,FA,KASE,PN)                            MARS0010
C                                                                      MARS0020
C     INPUTS ARE -- SNDB, SIGNAL-TO-NOISE POWER RATIO IN DECIBELS --   MARS0030
C     N, NUMBER OF PULSES INTEGRATED --                                MARS0040
C     FA, FALSE ALARM PROBABILITY, EXPRESSED AS ABSOLUTE VALUE OF POWER MARS0050
C     OF TEN (E.G., FA = 8. MEANS 10.**(-8.) FALSE ALARM PROBABILITY -- MARS0060
C     KASE, SWERLING FLUCTUATION MODEL, WITH KASE = 0 FOR NONFLUCTUATINGMARS0070
C     OUTPUT PN IS PROBABILITY OF DETECTION                            MARS0080
C                                                                      MARS0090
C     BASED ON PROGRAM WRITTEN ATJHU APPLIED PHYSICS LABORATORY, NAMED  MARS0100
C     SUBROUTINE MARCUM.  MODIFIED AT NRL BY L. V. BLAKE.  THIS VERSION MARS0110
C     DATED APRIL 1971                                                 MARS0120
C     APL VERSION DEFINED FA AS FALSE ALARM NUMBER (MARCUM CONCEPT).   MARS0130
C     NRL MOD CHANGED THIS TO FALSE ALARM PROBABILITY (AS DEFINED ABOVE)MARS0140
C     SOME OTHER CHANGES ALSO.                                         MARS0150
C                                                                      MARS0160
      DOUBLE PRECISION FAPR,YBPR,GAMPR,PYB,H,YO,EO,Y1,E1,STEP,YB       MARS0170
      DOUBLE PRECISION DGAM, DEVAL, SUMLOG, SJML, FAN, FA              MARS0180
C                                                                      MARS0190
C     COMPUTE MARCUM-SWERLING DETECTION PROBABILITIES                  MARS0200
C                                                                      MARS0210
      DATA NPREV,FAPREV/0,0./                                          MARS0220
C     CONVERT S/N IN DB TO NUMERICAL S/N RATIO                         MARS0230
C                                                                      MARS0240
      SNR = 10.**(SNDB*.1)                                             MARS0250
C                                                                      MARS0260
C     MODE=1                                                           MARS0270
C     IF MODE IS 1, CONVERT FA  TO MEAN EXPONENT OF FALSE-ALARM        MARS0280
C     PROBABILITY RATHER THAN MARCUM FALSE-ALARM NUMBER                MARS0290
C                                                                      MARS0300
C     IF (MODE) 800, 800, 900                                          MARS0310
  900 FAN = DLOG10(DLOG(.5D0)/DLOG(1.D0 - (10.D0)**(-FA)))             MARS0320
C     GO TO 905                                                        MARS0330
C 800 FAN = FA                                                         MARS0340
C                                                                      MARS0350
C     TEST INPUTS                                                      MARS0360
C                                                                      MARS0370
  905 IF(N) 99,99,2                                                    MARS0380
```

```
    2 IF(FA)99,99,3
    3 IF(KASE) 99,4,4
    4 IF(KASE-4) 5,5,99
C
C     ESTIMATE BIAS LEVEL
C
    5 ENPR = 0.
    6 ENPR = FAN
      EN = N
      YBPR = 0.
      IF (NPREV .EQ. N .AND. FAPREV .EQ. FA) GO TO 777
      IF(N-12) 7,7,8
    7 YBPR = EN*(1.D0 + 2.2D0*ENPR/EN**((2.D0/3.D0) + .015D0*ENPR))
      GO TO 11
    8 YBPR = EN*(1.+1.3*ENPR/EN**(.5+.011*ENPR))
C
C     COMPUTE BIAS LEVEL
C
   11 ENPR = 10.**ENPR
      GAMPR = DGAM(YBPR,N-1)
      PYB = .5**(1./ENPR)
      SUML = SUMLOG(N-1)
      IF(GAMPR-PYB) 10,12,12
   10 H=.1
      GO TO 14
   12 H=-.1
   14 Y0 = YBPR
      E0 = DEVAL(Y0,N-1,SUML)
   16 Y1 = Y0+H
      E1 = DEVAL(Y1,N-1,SUML)
      STEP = GAMPR + H*(E0+E1)/2.
      IF(DSIGN(1.D0,STEP-PYB)-DSIGN(1.D0,H)) 18,20,18
   18 Y0 = Y1
      E0 = E1
      GAMPR = STEP
      GO TO 16
   20 IF(H) 22,24,24
   22 YB = Y1 - H*(PYB-STEP)/(GAMPR-STEP)
      GO TO 30
   24 YB = Y0 + H*(PYB-GAMPR)/(STEP-GAMPR)
   30 BIAS = YB
  777 YB = BIAS
      Y0 = YB
      NPREV = N
      FAPREV = FA
C
C     SELECT M-S CASE
C
      X = SNR
      K = KASE+1
      GO TO (100,200,300,400,500), K
C
C     CASE 0
C
  100 SUM = 0.
```

```
      P = EN* X
      IF(YB-P-EN) 150,102,102
  102 KS = -(EN + 1.D0)/2.D0 + DSQRT(((EN-1.D0)/2.D0)**2 +P*YB)
      KS =  MAXO (KS,0)
      GS = 1.-GAM(YC,KS+N-1,TN)
      TS = EVAL(P,KS)*GS
      G = GS
      K = KS
      TERM = TS
      TL = TN
  110 TEMP = SUM+TERM
      IF(SUM-TEMP) 112,116,116
  112 SUM = TEMP
      IF(K) 116,116,114
  114 TERM = TERM*FLOAT (K)*(G-TL)/(P*G)
      G = G-TL
      K = K-1
      TL = TL*FLOAT (K+N)/YB
      GO TO 110
  116 TL = TN*YB/FLOAT (KS+N)
      K = KS+1
      G = GS+TL
      TERM = TS*P*G/(GS*FLOAT (K))
  120 TEMP = SUM+TERM
      IF(SUM-TEMP) 122,190,190
  122 SUM = TEMP
      TL = TL*YB/FLOAT (K+N)
      K = K+1
      TERM = TERM*P*(G+TL)/(G*FLOAT (K))
      G = G+TL
      GO TO 120
  150 KS = -1.D0 - EN/2.D0 + DSQRT(EN**2/4.D0 +P*YB)
      KS = MAXO (KS,0)
      GS = GAM(YC,KS+N-1,TN)
      IF(GS) 174,174,155
  155 TS = EVAL(P,KS)*GS
      G = GS
      TERM = TS
      K = KS
      TL = TN
  160 TEMP = SUM+TERM
      IF(SUM-TEMP) 162,166,165
  162 SUM = TEMP
      IF(K) 166,166,164
  164 TERM = TERM*FLOAT (K)*(G+TL)/(P*G)
      G = G+TL
      TL = TL*FLOAT (K+N-1)/YB
      K = K-1
      GO TO 160
  166 TL = TN*YB/FLOAT (KS+N)
      K = KS+1
      G = GS-TL
      TERM = TS*P*G/(GS*FLOAT (K))
  170 TEMP = SUM + TERM
      IF(SUM-TEMP) 172,174,174
  172 SUM = TEMP
      TL = TL*YB/FLOAT (K+N)
      TERM = TERM*P*(G-TL)/(G*FLOAT (K+1))
      G = G-TL
      K = K+1
      GO TO 170
  174 SUM = 1.-SUM
  190 PN = SUM
      GO TO 90
C
C     CASE  1
C
  200 IF(N-1) 210,210,220
  210 PN = DEXP(-YB/(1.DO +X))
      GO TO 90
  220 TEMP = 1. + 1./(EN*X)
      PN = 1.DO -GAM(YC,N-2,DUM) +DEXP((EN-1.DO)*ALOG(TEMP)
     1 -YB/(1.DO+EN *X))
     1                                  *GAM(YC/TEMP,N-2,DUM)
      GO TO 90
C
C     CASE  2
C
  300 IF(N-1) 310,310,320
  310 PN = DEXP(-YB/1.DO+X)
      GO TO 90
  320 PN = 1. - GAM(YC/(1.+X),N-1,DUM)
      GO TO 90
C
C     CASE 3
C
  400 IF(N-2) 410,420,430
  410 PN = (1.DO + 2.DO* X* YB/(X+2.DO)**2)*DEXP(-2.DO*YB/(2.DO+X))
      GO TO 90
  420 PN = (1.DO+YB/(1.DO+X))*DEXP(-YB/(1.DO+ X))
      GO TO 90
  430 C = 2./(2.+EN*X)
      D = 1.-C
      IF(YB*D-EN)  440,450,450
  440 SUM = 0.
      TERM = 1.
      J = N
  442 TEMP = SUM+TERM
      IF( SUM-TEMP ) 444,446,446
  444 SUM  = TEMP
      TERM = TERM*YB*D/FLOAT  (J)
      J = J+1
      GO TO 442
  446 PN = 1. - GAM(YC,N-2,DUM) + C*YB*EVAL(YC,N-2)
     1          + D*EVAL(YC,N-1)*(1.+C*YB-(EN-2.)*C/D)*SUM
      GO TO 90
  450 PN = 1.DO-GAM(YC,N-3,DUM) + YB*EVAL(YC,N-3)*C/D
     1   +DEXP(-C*YB-(EN-2.DO)*ALOG(D))*(1.DO+C*YB-(EN-2.DO)*C/D)
     2          *GAM(YC*D,N-3,DUM)
      GO TO 90
```

```
C
C     CASE 4
C
  500 SUM = 0.
      C = 2./(2.+X)
      D = 1.-C
      Q = C/D
      P = C*YB
      KS = (3.D0*EN +(YB*D))/2.D0-DSQRT((EN-1.D0+(YB*D))**2/4.D0
     1 +(YB*D)*(EN +1.D0))
      KS =   MIN0 (KS,N)
      KS =   MAX0 (KS,0)
      K = KS
      J = N-KS
      FKS = KS
      K =   MIN0 (KS,N)
      IF(YB-EN*(1.+D)) 550,501,501
  501 GS = 1. - GAM(P,2*N-1-KS,TN)
      IF(GS) 526,526,502
  502 TS = DEXP(FKS*ALOG(C)+(EN-FKS)*ALOG(D)+SUMLOG(N)-SUMLOG(KS)
     1            -SUMLOG(J)+ALOG(GS))
      G = GS
      TERM = TS
      TL = TN
  510 TEMP = SUM+TERM
      IF(SUM-TEMP) 512,516,516
  512 SUM = TEMP
      IF(K) 516,516,514
  514 TL = TL*P/FLOAT (2*N-K)
      TERM = TERM*FLOAT (K)*(G+TL)/(Q*FLOAT (N-K+1)*G)
      G = G+TL
      K = K-1
      GO TO 510
  516 IF(KS-N) 518,526,526
  518 TERM = TS*Q*FLOAT (N-KS)*(GS-TN)/(FLOAT (KS+1)*GS)
      G = GS-TN
      TL = TN*FLOAT (2*N-1-KS)/P
      K = KS+1
  520 TEMP = SUM+TERM
      IF(SUM-TEMP) 522,526,526
  522 SUM = TEMP
      IF(K-N) 524,526,526
  524 TERM = TERM*Q*FLOAT (N-K)*(G-TL)/(FLOAT (K+1)*G)
      G = G-TL
      TL = TL*FLOAT (2*N-1-K)/P
      K = K+1
      GO TO 520
  526 PN = SUM
      GO TO 90
  550 GS = GAM(P,2*N-1-KS,TN)
      IF(GS) 576,576,552
  552 TS = DEXP(FKS*ALOG(C)+(EN-FKS)*ALOG(D)+SUMLOG(N)-SUMLOG(KS)
     1            -SUMLOG(J)+ALOG(GS))
      G = GS
      TERM = TS
      TL = TN                                                           MARS2590
  560 TEMP = SUM+TERM                                                   MARS2600
      IF(SUM-TEMP) 562,566,566                                          MARS2610
  562 SUM = TEMP                                                        MARS2620
      IF(K) 566,566,564                                                 MARS2630
  564 TL = TL*P/FLOAT (2*N-K)                                           MARS2640
      TERM = TERM*FLOAT (K)*(G-TL)/(Q*FLOAT (N-K+1)*G)                  MARS2650
      G= G-TL                                                           MARS2660
      K = K-1                                                           MARS2670
      GO TO 560                                                         MARS2680
  566 IF(KS-N) 568,576,576                                              MARS2690
  568 TERM = TS*Q*FLOAT (N-KS)*(GS+TN)/(FLOAT (KS+1)*GS)                MARS2700
      G = GS+TN                                                         MARS2710
      TL = TN*FLOAT (2*N-1-KS)/P                                        MARS2720
      K = KS+1                                                          MARS2730
  570 TEMP = SUM+TERM                                                   MARS2740
      IF(SUM-TEMP) 572,576,576                                          MARS2750
  572 SUM = TEMP                                                        MARS2760
      IF(K-N) 574,576,576                                               MARS2770
  574 TERM = TERM*Q*FLOAT (N-K)*(G+TL)/(FLOAT (K+1)*G)                  MARS2780
      G= G+TL                                                           MARS2790
      TL = TL*FLOAT (2*N-1-K)/P                                         MARS2800
      K = K+1                                                           MARS2810
      GO TO 570                                                         MARS2820
  576 PN = 1.-SUM                                                       MARS2830
      GO TO 90                                                          MARS2840
C                                                                       MARS2850
C     SET PROBABILITY                                                   MARS2860
C                                                                       MARS2870
   90 IF(PN) 91,94,92                                                   MARS2880
   91 PN = 0.                                                           MARS2890
      GO TO 94                                                          MARS2900
   92 IF(PN-1.) 94,94,93                                                MARS2910
   93 PN = 1.                                                           MARS2920
   94 RETURN                                                            MARS2930
C                                                                       MARS2940
C     ERROR MESSAGE FOR BAD INPUTS                                      MARS2950
C                                                                       MARS2960
   99 WRITE (6,.9)  N,FA ,SNR,KASE                                      MARS2970
    9 FORMAT (1H0 /50H UNREASONABLE CALL SEQUENCE TO MARCUM, ZERO RESULTMARS2980
     1                7HS GIVEN //4H N = I8,5X,5HFA  = E16.8,5X,5HSNR = MARS2990
     2               E16.8,5X,6HKASE = I8)                              MARS3000
      PN = 0.                                                           MARS3010
      PIAS = 0.                                                         MARS3020
      RETURN                                                            MARS3030
      END                                                               MARS3040


      SUBROUTINE FGCSC(GNORM,ANG1,ANG2,PHORAD)                          FGCS0010
      PI2 = 1.5707963000                                                FGCS0020
      PHEE = ANG1 - ANG2                                                FGCS0030
      ANG3 = PHEE + PHORAD                                              FGCS0040
      IF(ANG3.GT.PHORAD) GO TO 10                                       FGCS0050
      T = SIN(PHORAD)                                                   FGCS0060
      BWR = .89*T                                                       FGCS0070
      X = (2.78345*SIN(PHEE))/BWR                                       FGCS0080
      AX = ABS(X)                                                       FGCS0090
      IF (AX .GE. .00001) GO TO 5                                       FGCS0100
      GNORM = 1.0                                                       FGCS0110
      GO TO 30                                                          FGCS0120
    5 GNORM = SIN(X)/X                                                  FGCS0130
      ABANG = ABS(PHEE)                                                 FGCS0140
      GO TO 20                                                          FGCS0150
   10 GNORM = SIN(PHORAD)/SIN(ANG3)                                     FGCS0160
      ABANG = ABS(ANG3)                                                 FGCS0170
   20 IF(ABANG.GT.PI2)  GNORM = .001                                    FGCS0180
   30 RETURN                                                            FGCS0190
      END                                                               FGCS0200


      FUNCTION DGAM(B,N)                                                DGAM0010
      DOUBLE PRECISION  SUM,  TERM,TEMP,FJ,DGAM, DEVAL, B, SUML,SUMLOG  DGAM0020
C           INTEGRAL = 1-(SUM, J=0 TO N, OF EXP (J*ALOG(B)-B-ALOG(NFACDGAM0030
      SUM = 0.                                                          DGAM0040
      K = B                                                             DGAM0050
      IF(K-N) 100,200,200                                               DGAM0060
  100 J = N+1                                                           DGAM0070
      SUML = SUMLOG(J)                                                  DGAM0080
      TERM = DEVAL(B,J,SUML)                                            DGAM0090
   10 TEMP = SUM+TERM                                                   DGAM0100
      IF(SUM-TEMP) 15,20,20                                             DGAM0110
   15 SUM = TEMP                                                        DGAM0120
      J = J+1                                                           DGAM0130
      FJ = J                                                            DGAM0140
      TERM = TERM*B/FJ                                                  DGAM0150
      GO TO 10                                                          DGAM0160
   20 DGAM = SUM                                                        DGAM0170
      RETURN                                                            DGAM0180
  200 J = N                                                             DGAM0190
      SUML = SUMLOG(J)                                                  DGAM0200
      TERM = DEVAL(B,J,SUML)                                            DGAM0210
   30 TEMP = SUM+TERM                                                   DGAM0220
      IF(SUM-TEMP) 35,40,40                                             DGAM0230
   35 SUM = TEMP                                                        DGAM0240
      IF(J-1) 40,36,36                                                  DGAM0250
   36 FJ = J                                                            DGAM0260
      TERM = TERM*FJ/B                                                  DGAM0270
      J = J-1                                                           DGAM0280
      GO TO 30                                                          DGAM0290
   40 DGAM = 1.-SUM                                                     DGAM0300
      RETURN                                                            DGAM0310
      END                                                               DGAM0320


      FUNCTION DEVAL (Y,N,SUML)                                         DEVL0010
      DOUBLE PRECISION  XPON,EN,DEVAL,   Y,SUML                         DEVL0020
      XPON = -Y                                                         DEVL0030
```

AD-A048 342   LTV AEROSPACE CORP DALLAS TEX VOUGHT SYSTEMS DIV   F/G 15/7
SEATIDE ANALYSIS PROCESS. VOLUME IID. NAVAL ENGAGEMENT MODEL (N--ETC(U)
FEB 75   DAAB09-72-C-0062
UNCLASSIFIED   VSD-00.1636-VOL-2D-REV-A   NL

3 of 3
ADA048 342

END
DATE
FILMED
2-78
DDC

```
      IF(N) 20,20,1C                                                    DEVL0040
   10 EN = N                                                            DEVL0050
      XPON = XPON+EN*DLOG(Y)-SUML                                       DEVL0060
      IF (XPON .LT. -290.) GO TO 30                                     DEVL0070
   20 DEVAL = DEXP(XPON)                                                DEVL0080
      GO TO 40                                                          DEVL0090
   30 DEVAL = 0.                                                        DEVL0100
   40 RETURN                                                            DEVL0110
      END                                                               DEVL0120


      FUNCTION GAM(B,N,TN)                                              GAM 0010
C                         SINGLE PRECISION VERSION OF DGAM              GAM 0020
      SUM = 0.                                                          GAM 0030
      K = B                                                             GAM 0040
      IF(K-N) 100,200,200                                               GAM 0050
  100 J = N+1                                                           GAM 0060
      TERM = EVAL(B,J)                                                  GAM 0070
      TN = TERM*FLOAT (J)/B                                             GAM 0080
   10 TEMP  = SUM+TERM                                                  GAM 0090
      IF(SUM-TEMP) 15,20,20                                             GAM 0100
   15 SUM = TEMP                                                        GAM 0110
      J = J+1                                                           GAM 0120
      FJ = J                                                            GAM 0130
      TERM = TERM*B/FJ                                                  GAM 0140
      GO TO 10                                                          GAM 0150
   20 GAM = SUM                                                         GAM 0160
      RETURN                                                            GAM 0170
  200 J = N                                                             GAM 0180
      TERM = FVAL(B,J)                                                  GAM 0190
      TN = TERM                                                         GAM 0200
   30 TEMP = SUM+TERM                                                   GAM 0210
      IF(SUM-TEMP) 35,40,40                                             GAM 0220
   35 SUM = TEMP                                                        GAM 0230
      IF(J-1) 40,36,36                                                  GAM 0240
   36 FJ = J                                                            GAM 0250
      TERM = TERM*FJ/B                                                  GAM 0260
      J = J-1                                                           GAM 0270
      GO TO 30                                                          GAM 0280
   40 GAM = 1.-SUM                                                      GAM 0290
      RETURN                                                            GAM 0300
      END                                                               GAM 0310


      FUNCTION FVAL(Y,N)                                                FVAL0010
      DOUBLE PRECISION SUMLOG                                           FVAL0020
      XPON = -Y                                                         FVAL0030
      IF(N) 20,20,10                                                    FVAL0040
                                                                        FVAL0050
   10 EN = N                                                            FVAL0060
      XPON = XPON+EN*ALOG(Y)-SUMLOG(N)                                  FVAL0070
      IF (XPON .LT. -290.) GO TO 30                                     FVAL0080
   20 FVAL = EXP (XPON)                                                 FVAL0090
      GO TO 40
   30 DEVAL = 0.                                                          EVAL0100
   40 RETURN                                                              EVAL0110
      END                                                                 EVAL0120


      FUNCTION SUMLOG(N)                                                  SMLG0010
      DOUBLE PRECISION A, B, SUMLOG                                       SMLG0020
      DIMENSION A(1000)                                                   SMLG0030
      DATA DUMA,DUMB/0.,0./                                               SMLG0040
      NMAX=1000                                                           SMLG0050
      IF(DUMA-DUMB) 20,10,20                                              SMLG0060
   10 DUMA = 1.                                                           SMLG0070
      DUMB = 0.                                                           SMLG0080
      NLAST = 1                                                           SMLG0090
      A(1) = 0.                                                           SMLG0100
   20 NN = IABS (N)                                                       SMLG0110
      IF(NN-1) 30,30,40                                                   SMLG0120
   30 SUMLOG = 0.                                                         SMLG0130
      RETURN                                                              SMLG0140
   40 IF(NN-NLAST) 50,50,60                                               SMLG0150
   50 SUMLOG = A(NN)                                                      SMLG0160
      RETURN                                                              SMLG0170
   60 K = NLAST+1                                                         SMLG0180
      IF(NN-NMAX) 70,70,80                                                SMLG0190
   70 DO 72 I=K,NN                                                        SMLG0200
   72 A(I) = A(I-1) + DLOG(DFLOAT(I))                                     SMLG0210
      NLAST = NN                                                          SMLG0220
      GO TO 50                                                            SMLG0230
   80 IF(NLAST-NMAX) 82,90,90                                             SMLG0240
   82 DO 84 I=K,NMAX                                                      SMLG0250
   84 A(I) = A(I-1) + DLOG(DFLOAT(I))                                     SMLG0260
      NLAST = NMAX                                                        SMLG0270
   90 B = A(NMAX)                                                         SMLG0280
      K = NMAX+1                                                          SMLG0290
      DO 92 I=K,NN                                                        SMLG0300
   92 B = B + DLOG(DFLOAT(I))                                             SMLG0310
      SUMLOG = B                                                          SMLG0320
      RETURN                                                              SMLG0330
      END                                                                 SMLG0340


      SUBROUTINE INVERT(XMIN,XMAX,X,FT)                                   NVRT0010
      COMMON /BLK2/  T, TT                                                NVRT0020
C   THIS SUBROUTINE FINDS VALUE OF X THAT RESULTS IN F(X) = FT, BY        NVRT0030
C   ITERATION BASED ON LINEAR INTERPOLATION/EXTRAPOLATION FROM PREVIOUS   NVRT0040
C   TWO TRIALS.  FUNCTION F MUST BE MONOTONIC.                            NVRT0050
      TEST = 10000.                                                       NVRT0060
      FD = FT                                                             NVRT0070
      IF (FT .EQ. 0.) FD = 1.                                             NVRT0080
      NOI = 1                                                             NVRT0090
      X = (XMAX + XMIN) / 2.                                              NVRT0100
      F1 = FSS(X, T, TT)                                                  NVRT0110
      X2 = X                                                              NVRT0120
      F2 = F1                                                           NVRT0130
      FN = F1                                                           NVRT0140
      GO TO 4                                                           NVRT0150
    1 F1 = ESS(X, T, TT)                                                NVRT0160
      IF (NOI .EQ. 15) GO TO 2                                          NVRT0170
    4 TEST1 = ABS((F1 - FT) / FD)                                       NVRT0180
      IF (TEST1 - TEST) 2, 2, 3                                         NVRT0190
    3 XM = X                                                            NVRT0200
      IF (NOI .GT. 1) GO TO 6                                           NVRT0210
      DELTA = (XMAX - XMIN) / 4.                                        NVRT0220
      IF (FT .LT. F1) DELTA = - DELTA                                   NVRT0230
      FMAX = ESS(XMAX, T, TT)                                           NVRT0240
      FMIN = ESS(XMIN, T, TT)                                           NVRT0250
      IF (FMAX .LT. FMIN) DELTA = - DELTA                               NVRT0260
      X = X + DELTA                                                     NVRT0270
      XN = XM                                                           NVRT0280
      NOI = 2                                                           NVRT0290
      GO TO 1                                                           NVRT0300
    6 X = (FT-FN)*(X-XN)/(F1-FN) + XN                                   NVRT0310
      NOI = NOI + 1                                                     NVRT0320
      IF (NOI - 3) 24, 21, 24                                           NVRT0330
   21 IF (ABS(F2-FT) - ABS(F1-FT)) 23, 23, 24                           NVRT0340
   23 XN = X2                                                           NVRT0350
      FN = F2                                                           NVRT0360
      GO TO 1                                                           NVRT0370
   24 XN = XM                                                           NVRT0380
      FN = F1                                                           NVRT0390
      GO TO 1                                                           NVRT0400
    2 RETURN                                                            NVRT0410
      END                                                               NVRT0420


      FUNCTION ESS(S1, T, TT)                                           ESS 0010
C     DEFINES PARAMETER S AS A FUNCTION OF S1 AND T.                    ESS 0020
C     (SEE RAD LAB SERIES VOL. 13, P115.)                               ESS 0030
      SS1 = S1 * S1                                                     ESS 0040
      TEMP1 = 1. - SS1                                                  ESS 0050
      TEMP1   =  TEMP1 * TEMP1                                          ESS 0060
      TEMP2 = 4. * SS1 * TT                                             ESS 0070
      TEMP3 = 2. * S1                                                   ESS 0080
      TEMP4 = 1. + T                                                    ESS 0090
      ESS = (S1 + (SQRT(TEMP1+ TEMP2) - 1. + SS1) / TEMP3) / TEMP4      ESS 0100
      RETURN                                                            ESS 0110
      END                                                               ESS 0120


      SUBROUTINE SEAREF(FMHZ,PSI,RHO,PHI)                               SERF0010
      IMPLICIT REAL*8(A-H,O-Z)                                          SERF0020
C  MOCS.121,440-470.                                 VER.2   11-25-74 SERF0030
C     THIS SUPROUTINE COMPUTES COMPLEX REFLECTION COEFFICIENT OF        SERF0040
C     SEA WATER, AS A FUNCTION OF FREQUENCY, FMHZ  GRAZIAG ANGLE,       SERF0050
C     PSI,(RADIANS)  AND WAVE POLARIZATION (IPOL = 1 FOR VERTICAL       SERF0060
C     AND IPOL = 2 FOR HORIZONTAL POLARIZATION).                        SERF0070
```

```
C     OUTPUT OF THIS ROUTINE IS MAGNITUDE RHO AND PHASE ANGLE              SERF0080
C     PHI(RADIANS) OF THE COMPLEX COEFFICIENT. COMPUTATION                 SERF0090
C     IS BASED UPON EQUATIONS (1) AND (2) OF RAD LAB SERIES                SERF0100
C     VOL. 13, PAGE 396. THE SUBROUTINE WAS WRITTEN BY L.V.                SERF0110
C     BLAKE, NRL CODE 5370 SEPT 1969.                                      SERF0120
      COMPLEX*16 EPSC, GAM, SQTERM, TERM, GAMCC, DIFF1, DIFF2              SERF0130
      COMPLEX*8 GAMS                                                       SERF0140
      DATA  FLAST/0.0/                                                     SERF0150
      SINPSI  =  DSIN(PSI)                                                 SERF0160
      CSPSI   =  DCOS(PSI) ** 2                                            SERF0170
      IF (FMHZ .EQ. FLAST) GO TO 200                                       SERF0180
C     IF SUBROUTINE HAS BEEN CALLED PREVIOUSLY DURING THE PROGRAM,         SERF0190
C     AND FREQUENCY IS THE SAME AS ON THE LAST PREVIOUS CALL,              SERF0200
C     PART OF THE COMPUTATION NEED NOT BE DONE, SINCE THE                  SERF0210
C     REQUIRED VALUES HAVE BEEN STORED.                                    SERF0220
      FLAST = FMHZ                                                         SERF0230
C COMPUTE WAVELENGTH                                                       SERF0240
      W       =  299.793D0 / FMHZ                                          SERF0250
      IF (FMHZ .GT. 1500.D0)  GO TO 151                                    SERF0260
C     SIG IS CONDUCTIVITY(MHO/METER), EPSI IS THE REAL PART OF             SERF0270
C     THE DIELECTRIC CONSTANT. BOTH ARE DEPENDENT UPON FMHZ                SERF0280
  150 SIG     =  4.3D0                                                     SERF0290
      EPS1    =  80.D0                                                     SERF0300
      GO TO 155                                                            SERF0310
  151 IF (FMHZ .GT. 3000.D0)  GO TO 154                                    SERF0320
      SIG     =  4.3D0 + (FMHZ - 1500.D0) * .00148D0                       SERF0330
  153 EPS1    =  80.D0 - (FMHZ - 1500.D0) * .00733D0                       SERF0340
      GO TO 155                                                            SERF0350
  154 EPS1    =  69.D0 - (FMHZ - 3000.D0) * .002429D0                      SERF0360
      SIG     =  6.52D0+ (FMHZ - 3000.D0) * .001314D0                      SERF0370
  155 EPSC    =  DCMPLX(EPS1,-60.D0*W*SIG)                                 SERF0380
  200 SQTERM  =  CDSQRT(EPSC  -  CSPSI)                                    SERF0390
C     IF (IPOL .GT. 1)          GO TO 161                                  SERF0400
C 160 TERM =     EPSC * SINPSI                                             SERF0410
C     GAM = (TERM-SQTERM)/(TERM+SQTERM)                                    SERF0420
C     GO TO 180                                                            SERF0430
  161 GAM = (SINPSI  -  SQTERM) / (SINPSI  +  SQTERM)                      SERF0440
  180 RHO     =  CDABS(GAM)                                                SERF0450
      GAMS = GAM                                                           SERF0460
      PHI = -ATAN2(AIMAG(GAMS), REAL(GAMS))                                SERF0470
C                                                                          SERF0480
C                                                                          SERF0490
      RETURN                                                               SERF0500
      END                                                                  SERF0510


      SUBROUTINE DFRACT (AHFT,THFT,RNM,FMHZ,FDB)                           DFRC0010
      IMPLICIT REAL*8(A-H,O-Z)                                             DFRC0020
      REAL*4 AHFT, THFT                                                    DFRC0030
C                                                                          DFRC0040
C     [illegible] IS ANTENNA HEIGHT(FEET)  THFT IS TARGET HEIGHT(FEET).    DFRC0050
C     [illegible] IS RANGE IN NAUTICAL MILES, FMHZ IS FREQUENCY IN         DFRC0060
C     [illegible]ERTZ.                                                     DFRC0070
C     THIS ROUTINE COMPUTES PROPAGATION FACTOR IN DB RELATIVE TO           DFRC0080
```

```
C       FREE SPACE. BASIS IS EQUATION  463 OF   PROPAGATION OF SHORT          DFRC0090
C       RADIO WAVES,  KERR, VOL 13 OF RAD LAB SERIES, PAGE 122                DFRC0100
C       Z1 AND Z2 ARE  NATURALHEIGHTS  AND X IS  NATURAL RANGE , SEE          DFRC0110
C       EQUATIONS 351 AND 358, PAGES 96-97, KERR.                             DFRC0120
C                                                                             DFRC0130
        PWR      =  1.D0 / 3.D0                                               DFRC0140
        FACTOR   = (FMHZ **(2.D0 * PWR)) / 6988.103D0                         DFRC0150
        Z1 = AHFT * FACTOR                                                    DFRC0160
        Z2 = THFT * FACTOR                                                    DFRC0170
        X        = (RNM  * (FMHZ ** PWR))/ 102.715D0                          DFRC0180
        CALL UFCN (Z1,UDB1)                                                   DFRC0190
        CALL UFCN (Z2,UDB2)                                                   DFRC0200
        FDB      = 10.992098640C + 10.D0*DLOG10(X) - 17.545497D0 *X           DFRC0210
     1             + UDB1  + UDB2                                             DFRC0220
        RETURN                                                                DFRC0230
        END                                                                   DFRC0240


        SUBROUTINE UFCN (Z, UDB)                                              UFCN0010
        IMPLICIT REAL*8(A-H,O-Z)                                              UFCN0020
C       THIS SUBROUTINE COMPUTES HEIGHT-GAIN FUNCTION,UDB, IN DECIBELS        UFCN0030
C       BY USING EMPIRICAL FORMULAS FOR DIFFERENT SEGMENTS OF FIG.            UFCN0040
C       2.20, PAGE 128 OF   PROPAGATION OF SHORT RADIO WAVES , KERR,          UFCN0050
C       VOL 13 OF RAD LAB SERIES.                                             UFCN0060
C       NOTE  THIS CURVE IS VALID FOR HORIZONTAL POLARIZATION ONLY.           UFCN0070
        IF (Z .GT. .6D0)          GO TO 2                                     UFCN0080
        UDB      =  20.D0 * DLOG10(Z)                                         UFCN0090
        RETURN                                                                UFCN0100
      2 IF (Z .GE. 1.D0)          GO TO 4                                     UFCN0110
        UDB      = -4.3D0 + 51.04D0*(DLOG10(Z/.6D0)) ** 1.4D0                 UFCN0120
        RETURN                                                                UFCN0130
      4 UDB      =  19.84728D0 *((Z ** .47D0) - .9D0)                         UFCN0140
        RETURN                                                                UFCN0150
        END                                                                   UFCN0160


        SUBROUTINE CURVIT                                                     CRVT0010
        IMPLICIT REAL*8(A-H,O-Z)                                              CRVT0020
C                                                                             CRVT0030
C THIS SUBROUTINE IS USED TO FIND THE COEFFICIENTS OF A CUBIC EQUATION        CRVT0040
C FOR USE IN CALCULATING PATTERN FACTORS IN THE INTERMEDIATE ZONE.            CRVT0050
C                                                                             CRVT0060
        COMMON /BLK1/    XX(4), YY(4), A, B, C, D                             CRVT0070
        DIMENSION        AR(4,4), AY(4,4)                                     CRVT0080
C                                                                             CRVT0090
        DO 5   I = 1,4                                                        CRVT0100
           AY(I,1)  =  YY(I)                                                  CRVT0110
           AY(I,2)  =  0.0D0                                                  CRVT0120
           AY(I,3)  =  0.0D0                                                  CRVT0130
           AY(I,4)  =  0.0D0                                                  CRVT0140
           AR(I,1)  =  1.0D0                                                  CRVT0150
      5 CONTINUE                                                              CRVT0160
C                                                                             CRVT0170
```

```
      DO 10  I = 1,4                                                    CRVT0180
      DO 10  J = 2,4                                                    CRVT0190
         K          =  J  -  1                                          CRVT0200
         AR(I,J)    =  AR(I,K)   *   XX(I)                              CRVT0210
   10 CONTINUE                                                          CRVT0220
C                                                                       CRVT0230
      CALL MATALG(AR,AY)                                                CRVT0240
C                                                                       CRVT0250
C SET  COEFFICIENTS                                                     CRVT0260
C                                                                       CRVT0270
      A          =  AY(4,1)                                             CRVT0280
      B          =  AY(3,1)                                             CRVT0290
      C          =  AY(2,1)                                             CRVT0300
      D          =  AY(1,1)                                             CRVT0310
C                                                                       CRVT0320
      RETURN                                                            CRVT0330
      END                                                               CRVT0340


      SUBROUTINE MATALG(A,X)                                            MTLG0010
      IMPLICIT REAL*8(A-H,O-Z)                                          MTLG0020
      DIMENSION A(4,4),X(4,4)                                           MTLG0030
C     IF(IDO) 1,2,1                                                     MTLG0040
C   1 DO 3 I=1,NR                                                       MTLG0050
C     DO 4 J=1,NR                                                       MTLG0060
C   4 X(I,J)=0.0                                                        MTLG0070
C   3 X(I,I)=1.0                                                        MTLG0080
C     NV=NR                                                             MTLG0090
    2 DET      =  1.D0                                                  MTLG0100
      NR1 = 3                                                           MTLG0110
      DO 5 K = 1, 3                                                     MTLG0120
      IR1=K+1                                                           MTLG0130
      PIVOT    =  0.0D0                                                 MTLG0140
      DO 6 I = K, 4                                                     MTLG0150
      Z        =  DABS(A(I,K))                                          MTLG0160
      IF(Z-PIVOT) 6,6,7                                                 MTLG0170
    7 PIVOT=Z                                                           MTLG0180
      IPR=I                                                             MTLG0190
    6 CONTINUE                                                          MTLG0200
      IF(PIVOT) 8,9,8                                                   MTLG0210
    9 DET      =  0.0D0                                                 MTLG0220
      RETURN                                                            MTLG0230
    8 IF(IPR-K) 10,11,10                                                MTLG0240
   10 DO 12 J = K, 4                                                    MTLG0250
      Z=A(IPR,J)                                                        MTLG0260
      A(IPR,J)=A(K,J)                                                   MTLG0270
   12 A(K,J)=Z                                                          MTLG0280
      Z = X(IPR,1)                                                      MTLG0290
      X(IPR,1) = X(K,1)                                                 MTLG0300
      X(K,1) = Z                                                        MTLG0310
      DET=-DET                                                          MTLG0320
   11 DET=DET*A(K,K)                                                    MTLG0330
      PIVOT    =  1.0D0 / A(K,K)                                        MTLG0340
      DO 14 J = IR1, 4                                                  MTLG0350
      A(K,J)=A(K,J)*PIVOT                                               MTLG0360
      DO 14 I = IP1, 4                                                  MTLG0370
   14 A(I,J)=A(I,J)-A(I,K)*A(K,J)                                       MTLG0380
      IF (X(K,1)) 15, 5, 15                                             MTLG0390
   15 X(K,1) = X(K,1) * PIVOT                                           MTLG0400
      DO 16 I = IP1, 4                                                  MTLG0410
   16 X(I,1) = X(I,1) - A(I,K) * X(K,1)                                 MTLG0420
    5 CONTINUE                                                          MTLG0430
      IF (A(4,4)) 17, 9, 17                                             MTLG0440
   17 DET = DET * A(4,4)                                                MTLG0450
      PIVOT = 1. / A(4,4)                                               MTLG0460
      X(4,1) = X(4,1) * PIVOT                                           MTLG0470
      DO 18 K = 1, 3                                                    MTLG0480
      I = 4 - K                                                         MTLG0490
      SUM     =   0.0D0                                                 MTLG0500
      DO 19 L = I, 3                                                    MTLG0510
   19 SUM = SUM + A(I,L+1)*X(L+1,1)                                     MTLG0520
   18 X(I,1) = X(I,1) - SUM                                             MTLG0530
      RETURN                                                            MTLG0540
      END                                                               MTLG0550


      SUBROUTINE ANGER(VAR,SNRI,ANG3,CON1,CON2)                         ANGR0010
      XNUM = ANG3*CON1                                                  ANGR0020
      DEN = CON2*SQRT(SNRI)                                             ANGR0030
      RMS = XNUM/DEN                                                    ANGR0040
      RMS = RMS*1000.                                                   ANGR0050
      VAR = RMS**2                                                      ANGR0060
      RETURN                                                            ANGR0070
      END                                                               ANGR0080


      SUBROUTINE FGAIN( GNORM , ANG1, ANG2, BWDR )                      FGAN0010
C     THIS SUBROUTINE COMPUTES NORMALIZED GAIN,GNORM, FOR SINX/X        FGAN0020
C       ANTENNA FOR VARIOUS ANGLES OFF BORESIGHT                        FGAN0030
C     THIS GAIN FUNCTION IS BASED ON A RECTANGULAR APERTURE WITH        FGAN0040
C       UNIFORM ILLUMINATION (BARTON &WARD,PAGE 24)                     FGAN0050
      PI2 = 1.5707963                                                   FGAN0060
      PHEE = ANG1 - ANG2                                                FGAN0070
      X = (2.783451*SIN(PHEE))/BWDR                                     FGAN0080
      AX = ABS(X)                                                       FGAN0090
      IF(AX.LT.1.E-5) GO TO 10                                          FGAN0100
      GNORM =  SIN(X)/X                                                 FGAN0110
      ABPHEE = ABS(PHEE)                                                FGAN0120
      IF(ABPHEE.GT.PI2)  GNORM = .001                                   FGAN0130
      RETURN                                                            FGAN0140
   10 GNORM = 1.0                                                       FGAN0150
      RETURN                                                            FGAN0160
      END                                                               FGAN0170


      SUBROUTINE SIGOS(ISS,XLAMDA,PSI,SIGODB)                           SIG 0010
      DIMENSION STOR(8,9,6)
      DIMENSION S1(8,9),S2(8,9),S3(8,9),S4(8,9),S5(8,9),S6(8,9)
      EQUIVALENCE (STOR(1,1,1),S1(1)), (STOR(1,1,2),S2(1)),
     1 (STOR(1,1,3),S3(1)),(STOR(1,1,4),S4(1)),(STOR(1,1,5),S5(1)),
     2 (STOR(1,1,6),S6(1))
      DATA S1 / 809.,1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-62.,-80.,-87.,-90.,-108.,-99.,-103.,
     2 .0052359,-56.,-73.,-79.,-83.,-95.,-93.,-97.,
     3 .0174532,-49.,-66.,-70.,-73.,-80.,-86.,-90.,
     4 .0523598,-49.,-58.,-68.,-68.,-72.,-75.,-79.,
     5 .1745329,-41.,-56.,-67.,-67.,-60.,-62.,-66.,
     6 .5235987,-32.,-57.,-62.,-52.,-50.,-49.,-53.,
     7 1.0471975,-22.,-34.,-25.,-32.,-32.,-32.,-36.,
     8 1.5707962,-18.,-11.,-3.,-7.,-10.,-15.,-19./
      DATA S2 / 809.,1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-58.,-68.,-75.,-80.,-99.,-97.,-101.,
     2 .0052359,-52.,-59.,-71.,-74.,-87.,-91.,-95.,
     3 .0174532,-45.,-51.,-56.,-65.,-73.,-84.,-88.,
     4 .0523598,-45.,-48.,-54.,-59.,-62.,-70.,-74.,
     5 .1745329,-38.,-51.,-53.,-58.,-56.,-57.,-61.,
     6 .5235987,-30.,-51.,-48.,-44.,-46.,-45.,-49.,
     7 1.0471975,-20.,-26.,-26.,-25.,-24.,-22.,-26.,
     8 1.5707962,-16.,-11.,-2.,-5.,-10.,-11.,-15./
      DATA S3/ 809.,1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-54.,-61.,-67.,-75.,-90.,-95.,-99.,
     2 .0052359,-48.,-53.,-60.,-66.,-78.,-89.,-93.,
     3 .0174532,-41.,-46.,-48.,-55.,-65.,-82.,-86.,
     4 .0523598,-41.,-42.,-48.,-53.,-59.,-66.,-70.,
     5 .1745329,-35.,-43.,-46.,-51.,-53.,-53.,-57.,
     6 .5235987,-28.,-44.,-42.,-40.,-41.,-42.,-46.,
     7 1.0471975,-19.,-23.,-22.,-23.,-21.,-22.,-26.,
     8 1.5707962,-15.,-11.,-1.,-4.,-9.,-11.,-15./
      DATA S4/ 809., 1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-50.,-53.,-60.,-68.,-82.,-90.,-94.,
     2 .0052359,-44.,-46.,-50.,-58.,-72.,-82.,-86.,
     3 .0174532,-37.,-40.,-43.,-48.,-60.,-76.,-80.,
     4 .0523598,-37.,-39.,-42.,-46.,-55.,-61.,-65.,
     5 .1745329,-33.,-37.,-40.,-46.,-48.,-50.,-54.,
     6 .5235987,-27.,-34.,-37.,-38.,-39.,-40.,-44.,
     7 1.0471975,-17.,-21.,-20.,-22.,-20.,-21.,-25.,
     8 1.5707962,-13.,-11.,-1.,-4.,-8. 7,-11.,-15./
      DATA S5/ 809.,1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-47.,-48.,-55.,-58.,-73.,-88.,-92.,
     2 .0052359,-41.,-42.,-45.,-50.,-63.,-78.,-82.,
     3 .0174532,-34.,-36.,-39.,-42.,-52.,-69.,-53.,
     4 .0523598,-34.,-35.,-38.,-41.,-48.,-57.,-61.,
     5 .1745329,-31.,-34.,-36.,-41.,-45.,-48.,-52.,
     6 .5235987,-24.,-33.,-35.,-37.,-37.,-39.,-43.,
     7 1.0471975,-15.,-20.,-15.,-21.,-18.,-21.,-25.,
     8 1.5707962,-11.,-11.,0.,-2.,-8.,-11.,-15./
      DATA S6/809.,1.765,3.226,5.357,10.,24.,60.,150.,
     1 .0017453,-44.,-42.,-48.,-53.,-65.,-84.,-88.,
     2 .0052359,-38.,-39.,-41.,-44.,-58.,-75.,-79.,
     3 .0174532,-31.,-33.,-35.,-42.,-50.,-65.,-69.,
     4 .0523598,-30.,-32.,-34.,-37.,-46.,-53.,-57.,
     5 .1745329,-29.,-31.,-33.,-38.,-43.,-46.,-50.,                     SIG 0570
     6 .5235987,-22.,-30.,-33.,-32.,-34.,-38.,-42.,                     SIG 0580
     7 1.0471975,-8.,-19.,-13.,-20.,-18.,-20.,-24.,                     SIG 0590
     8 1.5707962,-9.,-11.,+1.,-1.,-8.,-11.,-15./                        SIG 0600
      JZ = ISS + 1                                                      SIG 0610
      CALL TLU23(STOR,8,9,6,JZ,XLAMDA,PSI,SIGOCB,INDY,INDX,INY,INX)     SIG 0620
C     IF(INY.NE.0.OR.INX.NE.0) STOP                                     SIG 0630
      RETURN                                                            SIG 0640
      END                                                               SIG 0650
```

## 3.0 SUBPROGRAM FUNCTIONAL INDEX

The following index references each subprogram in the same order as the appendix listing. Comments added pertain to the function of subprograms.

| Page | Subprogram | Comments |
|---|---|---|
| N-5 | MAIN | NEM Executive main program; controls all functions. |
| N-7 | PAGE | Pages all output with page numbers, classification, etc. |
| N-8 | WCBD11 | Writes basic tables on disk 11. |
| N-10 | WMAT3 | Writes matrices. |
| N-10 | WRITRX | Writes matrices. |
| N-11 | SORT | Rank orders a list of values from smallest to largest. |
| N-11 | KORDER | Orders an array of integers from smallest to largest. |
| N-12 | TBASIC | Basic table processing executive routine. |
| N-16 | WBASIC | Writes basic tables. |
| N-17 | UPDA | Namelist update of basic tables. |
| N-20 | STRING | Manipulates strings of data on a sequential data set. |
| N-22 | CARD | Read and list input data cards. |
| N-22 | UNIQUE | Lists the unique values found in a general array of values. |
| N-23 | SETUPA | Sets up group routes from basic table inputs. |
| N-28 | NAVIT | Sets up group routes and initial positions. |
| N-30 | UNISET | Sets up units from basic table inputs. |
| N-32 | SETUPB | Sets up platforms vs subsystems from basic table inputs. |
| N-37 | SETUPC | Sets up ASM-SSM subsystem data from basic table inputs. |
| N-45 | STTRAJ | Sets up missile trajectory velocity-altitude profiles. |
| N-46 | SETUPD | Sets up misc. weapon & SAM subsystem data. |
| N-54 | SETUPE | Sets up Radar, Sonar and Jammer subsystem data. |
| N-61 | SETUPF | Sets up cruise missile homing radars and other general data. |
| N-65 | UCHEK | Utility driver for checking Navigation and Radar models. |
| N-65 | RADCK | NEM Radar model checker. |

| Page | Subprogram | Comments |
|---|---|---|
| N-67 | XECUTE | Begins engagement model. Defines and allocates dynamic storage. |
| N-69 | SETDS | Dynamic storage monitor. |
| N-72 | ENGAGE | Engagement executive routine. |
| N-77 | TILOOP | Engagement logic. |
| N-81 | HVTGT | High value target designation and priorities. |
| N-82 | GCDATA | Initializes group center data. |
| N-83 | SONARM | Computes ranges for sonar systems. |
| N-83 | SYSTEM | Sets up subsystems for each unit prior to engagement. |
| N-92 | SYSTST | Tests for specific subsystem in platform-subsystem matrix. |
| N-93 | ALLXYZ | Computes and outputs position, velocity and status for units on routes. |
| N-96 | BLUXYZ | Interpolates position-velocity of group centers. |
| N-98 | GRPMOV | Fast interpolation of group center position-velocity. |
| N-99 | TUXYZ | Computes position-velocity of units on intercept trajectories. |
| N-102 | GRTEST | Computes time for a specified range between units. |
| N-102 | GUID | Computes guidance errors. |
| N-104 | INCEPT | Computes time and position of intercept between units. |
| N-105 | RELATE | Determines interaction intervals between units. |
| N-109 | NQUIRE | Checks the information matrix for existing data between units. |
| N-110 | ADDSEG | Adds data to the information matrix. |
| N-111 | UPSEG | Updates existing data in the information matrix. |
| N-111 | DETECT | Governs detection between units. Calls Radar Model. |
| N-117 | SHIPXS | Adjusts radar cross-section of ships for horizon masking. |
| N-118 | ALLOCA | Governs weapon allocation between units. |
| N-132 | SAMLT | Computes launch and flight times for SAM intercepts. |
| N-138 | CONINT | Interpolates SAM launch envelope data. |
| N-139 | WPNTRM | Governs weapon termination of SAM's, torpedoes, ASW, etc. |

| Page | Subprogram | Comments |
|---|---|---|
| N-141 | KILLEX | Evaluates terminal effects for all weapon-target combinations. |
| N-151 | KILLHE | Evaluates ship damage due to HE and nuclear weapons. |
| N-153 | DELSEG | Deletes obsolete data from the information matrix. |
| N-153 | STATE | Computes the state code for ship targets. |
| N-154 | URN | Uniformly distributed random number generators. |
| N-154 | XNRN | Normally distributed random number generator. |
| N-154 | ACMTRM | Governs termination of attack A/C and cruise missiles. |
| N-156 | TLU23 | Table look-up interpolation utility. |
| N-157 | WRAD | Computes weapon radii for nuclear weapons. |
| N-159 | STATUS | Updates engagement status at end of each time step. |
| N-162 | USTAT | Updates unit status at end of each time step. |
| N-164 | SNAP | Outputs position, velocity and status for units on intercept trajectories. |
| N-166 | SUMOUT | Outputs engagement results in summary format. |
| N-168 | BLOCK DATA-A | Data initialization, default values. |
| N-169 | BLOCK DATA-B | Data initialization, default values. |
| N-170 | RADAR | Radar model executive routine. |
| N-177 | FFACT | Pattern propagation factor due to multipath and sidelobe. |
| N-181 | MARSWR | Single look probability of detection using Swerling Case No., S/N, number of pulses integrated, and false alarm exponent. |
| N-186 | FGCSC | Cosecant squared antenna gain vs. angle off boresight. |
| N-187 | DGAM | Function supporting MARSWR. |
| N-187 | DEVAL | " " " |
| N-188 | GAM | " " " |
| N-188 | EVAL | " " " |
| N-189 | SUMLOG | " " " |

| Page | Subprogram | Comments |
|---|---|---|
| N-189 | INVERT | Function supporting FFACT. |
| N-190 | ESS | " " " |
| N-190 | SEAREF | " " " |
| N-191 | DFRACT | " " " |
| N-192 | UFCN | " " " |
| N-192 | CURVIT | " " " |
| N-193 | MATALG | " " " |
| N-194 | ANGER | Variance of track angle error. |
| N-194 | FGAIN | Sin x/x antenna gain vs. angle off boresight. |
| N-194 | SIGOS | Backscatter coefficient of sea clutter in db. |

## 4.0 SUBPROGRAM ALPHABETICAL INDEX